Z MIKROKOMPUTEREM NA TY

5–6

Bajtek

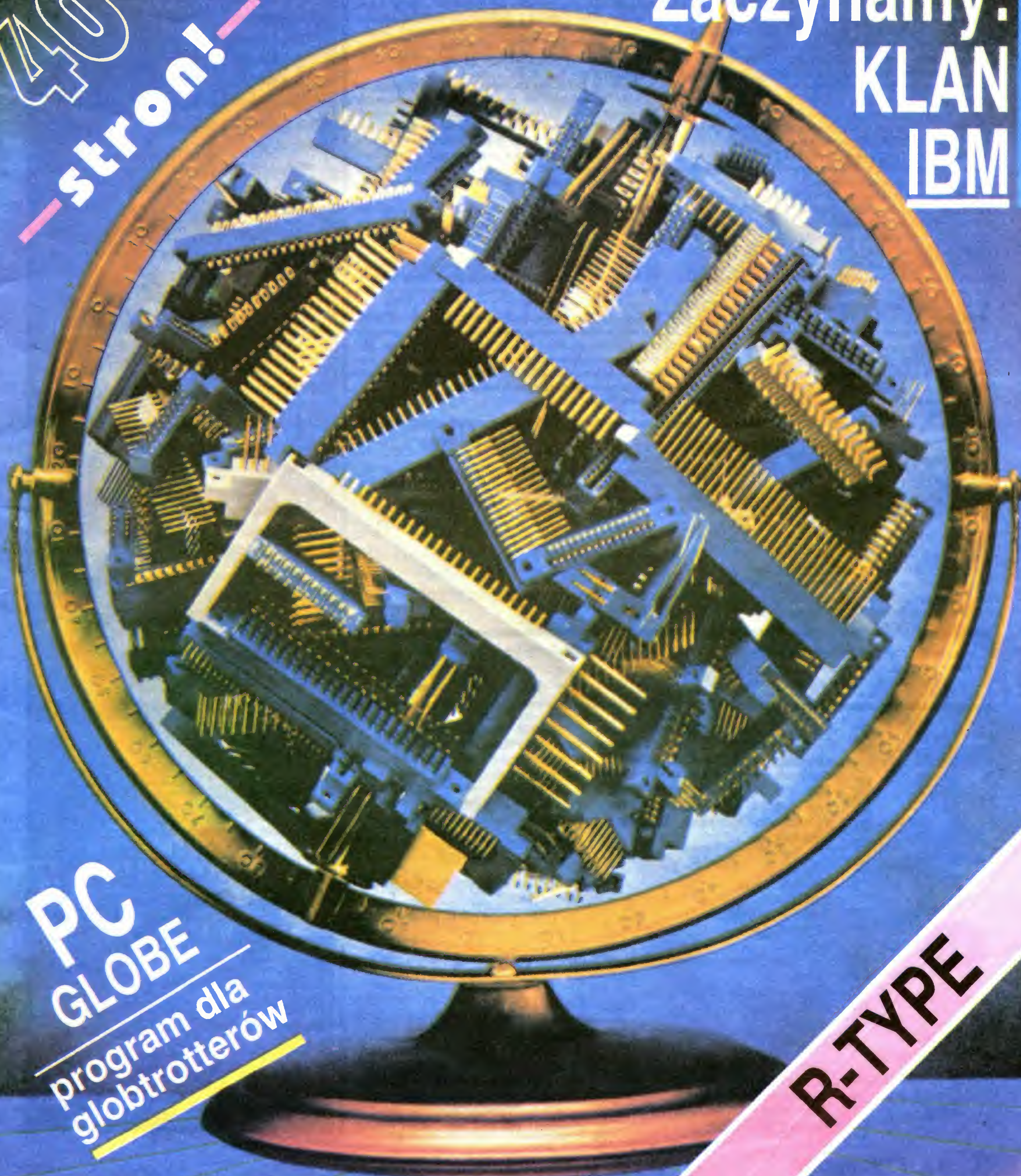
NR INDEKSU 353965
PL ISSN 0860-1674

MAGAZYN KOMPUTEROWY

NR I(51-52)'90 CENA 5000

40 stron!

Zaczynamy: KLAN IBM



PC GLOBE program dla globtrotterów

R-TYPE

UMARŁ KRÓL, NIECH ŻYJE KRÓL!

„Bajtek", tak jak prawie każdy tytuł, przeżył wstrząs. Było to coś w rodzaju zawału i wiązało się z dobrze każdemu znanym przełomem ekonomicznym w naszym kraju. Ceny papieru, usług drukarskich, haracz za kolportaż, brak pieniędzy — te rzeczy nie interesują Ciebie, Czytelniku i jest to zrozumiałe.

Winny jest nam znany. To stary żółw z literami „RSW" na twardej skorupie. Tony papieru zapisano już dywagacjami na temat działalności tego organu. Ale to też Ciebie nie interesuje.

Interesuje Cię natomiast efekt. To znaczy terminowe pojawianie się pisma w kioskach, jego poziom, aktualność itp. Tak, jak na to drugie wpływ ma prawie wyłącznie redakcja, tak pierwsze zależy jedynie od wydawcy. A on nie wywiązywał się ze swoich obowiązków.

Jak wiadomo, RSW jest w likwidacji. Razem z nią wszystkie podlegające jej tytuły, wydawnictwa, drukarnie i papiernie. Ostatecznym celem jest rozdzielenie majątku — rewindykacje, przekazania, wyprzedaż. W tej sytuacji, która była zapowiadana do lutego 1990, należało działać. I my to uczyniliśmy.

Na mocy Ustawy o Likwidacji RSW utworzyliśmy Spółdzielnię „Bajtek", w której skład weszły prawie wszystkie osoby zatrudnione w redakcji. Zarząd Spółdzielni wystąpił do Komisji Likwidacyjnej o przyznanie tytułu i części majątku. Jak na razie, tytuł został nam wydzierżawiony do momentu ostatecznej regulacji prawnej jego własności. Spółdzielnia zdobyła pieniądze na rozpoczęcie działalności i ruszyła pełną parą.

To, co widzisz przed sobą, tzn. numer 5–6 „Bajtka", jest pierwszym periodycznym numerem wydanym przez Spółdzielnię. Od dnia dzisiejszego regularne numery będą ukazywać się średnio co miesiąc.

Zauważyłeś chyba, że zwiększyliśmy objętość. „Bajtek" ma teraz 40 kolumn i planujemy dalsze tycie. Udało się nawet nie podnieść drastycznie ceny. Wypadek z ceną numeru 3–4, wydanym jeszcze przez Oficynę Wydawniczą „Sztandar Młodych" był próbą ratowania przez nią opróżnionej bezsensowną polityką kieszeni.

To nie jedyne nowum. Oprócz zwykłego „Bajtka", co miesiąc wychodzić będzie jeden numer specjalny. Wydawca, Spółdzielnia „Bajtek" posiada również prawa do tytułów „Moje Atari" oraz „Top Secret". Do końca roku planujemy ukazanie się po kilka numerów tych pism. W zamiarach leży wydanie jedno- lub kilkakrotnie numeru specjalnego poświęconego tylko Spectrum.

Mówimy jak dotąd o ilości, czas zabrać się za jakość. Na pewno nie ulegnie ona pogorszeniu. Zdajemy sobie sprawę, że w przeszłości popełniliśmy kilka błędów, ale uczynimy wszystko, by ich nie powtarzać. Zamierzamy stworzyć pismo mocne, oparte na rzeczowych i przydatnych informacjach. Nie będziemy stronić od publicystyki, solidnie weźmiemy się za edukację i popularyzację.

Jeśli chodzi o pozostałe dwa tytuły, to są one jakby kontynuacją wydawanych w przeszłości „Tylko o Atari" i „Gier".

Nie zrywamy żadnych kontaktów, umów itp. W kwestii kontaktów pismo-Czytelnik wszystko zostaje po staremu. Wszak to ci sami ludzie, ale już z nieco inną mocą decyzyjną co do spraw wydawniczych.

Zapraszamy do współpracy wszystkich, którzy chcieliby nas wspomóc. Już w tym numerze znajdziecie nowy klan — Klan IBM, który będzie się powiększał. Chcielibyśmy, aby wszyscy byli z nas zadowoleni. Naszym zadaniem jest zrobić wszystko w tym kierunku.

Redakcja

Prenumerata „Bajtka"

Ponownie otwieramy prenumeratę „Bajtka". Przyjmujemy wpłaty na konto Spółdzielnia „Bajtek", Bank „Agrobank S.A." 479994-1834-131, 04-398 Warszawa ul. Grochowska 262.

Trzy numery pisma 7/8, 9/10, 11/12 koszt — 15 000 zł

Jednocześnie polecamy prenumeratę naszych pozostałych tytułów „Moje Atari" i „Top Secret". Do końca roku wydamy po trzy numery każdego z tych pism. „Moje Atari" będzie kosztowało 4 500 zł a „Top Secret" — 5 000 zł. Wpłat prosimy dokonywać na to samo konto.

Wśród prenumeratorów w listopadzie rozlosujemy jako nagrodę komputer Commodore Amiga 500!

Zaufajcie nam — WŁAŚCICIELE PRACUJĄ LEPIEJ!

Z żalem informujemy, że nasz redakcyjny kolega Wojciech Zientara odpływając ze Szczecina na morza południowe zabrał ze sobą warunki zapowiadanego konkursu. Opublikujemy je natychmiast po powrocie sławnego żeglarza, któremu życzymy stopy wody pod kilem!

Bajtek MAGAZYN KOMPUTEROWY

Redakcja:
ul. Wspólna 61
00-687 Warszawa
Tel. 21-12-05

Redaguje Kolegium:
Marcin Borkowski, Marek Czarkowski, Łukasz Czekajewski, Janusz Jarmoch, Jarosław Młodzki, Waldemar Nowak, Maciej Pietraś, Marcin Przasnyski, Wanda Roszkowska, Wojciech Zientara

Szefowie Klanów:
Amstrad — Jonasz Mayer
Atari — Wojciech Zientara
Commodore — Klaudiusz Dybowski
Co Jest Grane — Marcin Przasnyski
IBM — Marcin Borkowski
Micro Magazyn — Janusz Jarmoch
Spectrum — Maciej Pietraś

Stali współpracownicy:
Marcin Bójko
Jarosław Burczyński
Piotr Kos
Robert Magdziak
Grzegorz Ostapiuk
Andrzej Pilaszek
Mieczysław Płacheta
Marek Sawicki
Piotr Sumara
Tomasz Tarczyński
Stanisław Winiecki

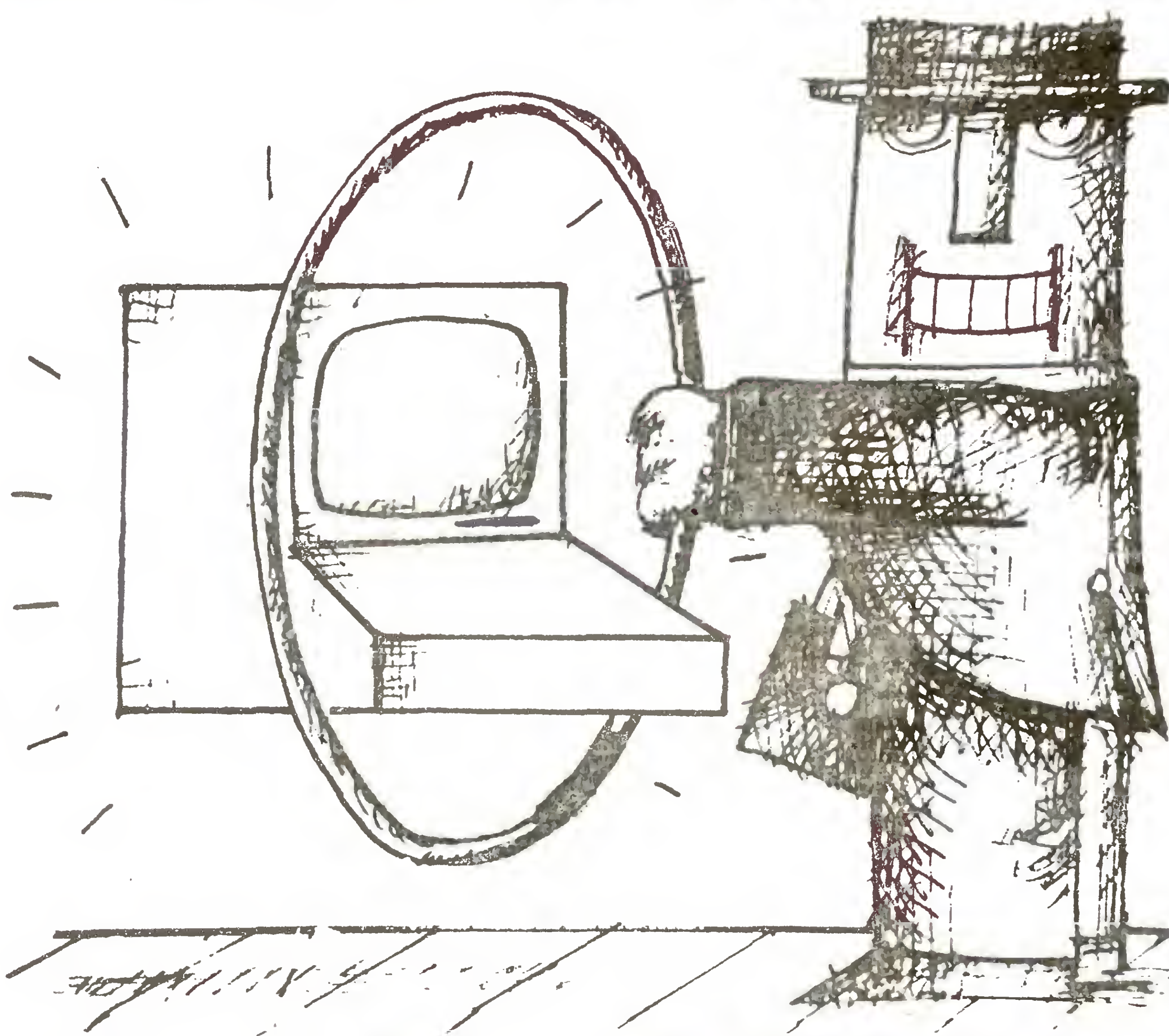
Opracowanie graficzne:
Wanda Roszkowska

Redaktor techniczny: Maria Radzimińska
Zdjęcia: Leopold Dzikowski

Wydawca:
Spółdzielnia „Bajtek"
ul. Wspólna 61
00-687 Warszawa

Skład i Druk:
Prasowe Zakłady Graficzne w Ciechanowie
Fotoskład: Grażyna Kurzątkowska
Montaż: Grażyna Ostaszewska
Korekta: Maria Krajewska
Teresa Rutkowska
Nakład: 105 tys. egz. Zam. 49610





Metody PANA B.

(poradnik warsztatowy ilustrowany przykładami)

Rosnące zainteresowanie komputerami powoduje wzrost zapotrzebowania na książki poświęcone informatyce. Aby ułatwić naszym czytelnikom zapełnianie luki powstałej na rynku księgarskim, proponujemy krótki kurs szybkiego pisania takich książek.

NA CZYM MOŻNA ZAROBIĆ?

Trudno operować konkretnymi liczbami, ale nie ulega wątpliwości, że nasz zysk jest wprost proporcjonalny do grubości książki, którą kupi od nas wydawnictwo. **Dlatego lepiej pisać książki grube niż cienkie.**

OD CZEGO ZACZĄĆ?

Choć bardziej opłaca się pisać książki grube, zaczynamy **od napisania książki cienkiej**, najlepiej na temat jakiejś wczesnej wersji programu lub systemu, którym się zajmiemy. Cienką książkę najwygodniej jest pisać opierając się na oryginalnej instrukcji do programu (może być ksero). Takie podejście do zagadnienia ma olbrzymią zaletę — za każdym razem, kiedy ukaże się nowa wersja programu (o co dba bez naszego udziału producent), możemy wykorzystać napisany już tekst jako podstawę nowej, grubszej książki. Zainteresowanych szczegółami tej techniki odsyłam do książek [2, 3], strony czternaste i większość pozostałych. Tego typu operacje najwygodniej przeprowadza się przy pomocy edytora tekstu (może być ChiWriter).

NASTĘPNY KROK

Kiedy mamy już gotową cienką książkę, zaczynamy ją pogrubiać. Można w tym celu zastosować kilka różnych technik, które pokrótce omówimy.

a) **Import** — przeglądamy wszystkie cienkie książki napisane przez nas wcześniej na dowolne tematy, w poszukiwaniu fragmentów nadających się do ponownego wykorzystania. Przykładowo, podczas pisania książki o kompilatorze języka C na IBM PC, możemy z powodzeniem wykorzystać fragmenty napisanej wcześniej książki o kompilatorze C na Spectrum [6 str. 25] i [5 str. 37].

b) **Programy przykładowe** — wszędzie tam, gdzie jest to możliwe, umieszczamy w tekście programy przykładowe. Na przykład, omawiając procedurę **circle** wprowadzamy w tekst zawierający kilkanaście linii program odwołujący się do tej procedury, obdarzając go stosownym komentarzem — „program rysuje kółko" [2 str. 175]. Można dodatkowo zamieścić wydruk ekranu — patrz podpunkt e.

c) **Powtarzanie wyliczeń** — za każdym razem, gdy nadarzy się okazja, wyliczamy wszystkie opcje, komunikaty, predefiniowane stałe, tłumaczenia kodów błędów itd. Wyliczenie wystarczy wpisać jednokrotnie, a potem kopiować blok tekstu [2 str. 188, 221, 249 i 316]. Nie umieszczając informacji o wyliczeniach w indeksie, zmniejszamy ryzyko wykrycia tego manewru.

d) **Programy demonstracyjne** — dołączane przez producenta dokumenty i programy demonstracyjne (znajdujące się na firmowych dyskietkach) dołączamy do naszej książki. Po skasowaniu tej ich części, której tłumaczenie wykorzystaliśmy podczas pisania początkowej, cienkiej wersji książki, resztę zostawiamy, dokonując w niej niewielkich zmian (niekoniecznie) i opatrując komentarzem „opracowano na podstawie ..." [4 str. 679].

e) **Wydruki ekranu** — po losowym wybraniu dowolnej opcji standardowo dostępnej w opisywanym programie naciskamy klawisz PrtScr (Print Screen). Otrzymany wydruk wmontowujemy do książki. Na jedną stronę wchodzą dwa takie wydruki [3 str. 612—625], albo jeden obrócony o dziewięćdziesiąt stopni [6 str. 106]. Można również wykorzystać rysunki robione przez program demonstracyjny przygotowany przez producenta (najlepiej wszystkie) [3 str. 685—711]. Uwaga — brak koloru na wydrukach nie jest powodem do rezygnacji z ich wykorzystania! [3 str. 689 i 691].

f) **Dodatki** — na końcu każdej książki poświęconej szeroko rozumianej informatyce, można zamieścić dodatek pod tytułem „Kody ASCII". Dobrze przygotowany dodatek moze służyć latami [2, 3]. Innymi dodatkami nadającymi się do wielokrotnego wykorzystania są między innymi: słownik terminów [4, 6], opis edytora [1, 2, 3], wykazy błędów [1, 2, 3] (niezależnie od wyliczeń w tekście — patrz punkt c) itd.. Przedstawionymi dodatkami można pogrubić każdą książkę [5].

UWAGI TECHNICZNE

Żeby przyspieszyć wydanie książki, a tym samym moment, w którym wszystkie pieniądze trafią do naszej kieszeni, możemy zaproponować wydawnictwu własny skład i druk pierwowzoru. Dzięki temu książka nie będzie przepisywana ani składana w drukarni, która ograniczy się do przefotografowania naszych wydruków [1, 2, 3]. Skracamy tym samym znacznie cykl wydawniczy, czyli przyspieszamy moment ujrzenia pieniędzy.

Podczas kopiowania bloków z jednego miejsca książki na drugie, należy zwrócić szczególną uwagę na konieczność dokonywania w nich drobnych zmian w zależności od kontekstu. Przykładem (niestety negatywnym) braku takiego działania jest książka [3], w której opis rozkazu LODSW na stronie 534, skopiowany ze strony 533, zawiera błąd, gdyż rozkaz dotyczy nie rejestru AL, a rejestru AX, i nie bajtu, a słowa.

WYDAJNOŚĆ

Stosując łącznie wszystkie opisane techniki, można w ciągu dwóch lat z książki [1] o objętości początkowej 168 stron (czyli dziesięć arkuszy wydawniczych), uzyskać z jednym etapem pośrednim — [2] o objętości 416 stron (23 arkusze) — efekt końcowy 752 stron (arkuszy wydawniczych 35). Równocześnie z przygotowanych już w trakcie pisania wszystkich trzech książek materiałów jesteśmy w stanie złożyć w kilka dni nową książkę na dowolnie obrany temat.

OSTATNI ETAP

Polega na znalezieniu wydawcy, który albo nie zna się na informatyce, albo nie czytał naszych poprzednich książek, albo ma do swoich czytelników taki sam stosunek, jak my do niego.

Po oddaniu wydawnictwu nowej książki i zainkasowaniu pieniędzy, możemy robić różne rzeczy, w szczególności czekać na ukazanie się nowej, jeszcze lepszej wersji opisanego programu, będącej okazją do napisania nowej, jeszcze grubszej książki.

Marcin Borkowski



[1] Jan Bielecki, Turbo Pascal wersja 3.0, WNT, Warszawa 1987
[2] Jan Bielecki, Rozszerzony Turbo Pascal wersja 4.0. WKiŁ, Warszawa 1988
[3] Jan Bielecki, Turbo Pascal 5.0 wersja profesjonalna, WKiŁ, Warszawa 1989
[4] Jan Bielecki, Język C interpretacja standardu, WNT, Warszawa 1988
[5] Jan Bielecki, Sam na sam z językiem C, WKiŁ, Warszawa 1988
[6] Jan Bielecki, Turbo C dla programistów, WKiŁ, Warszawa 1989







ODPOWIEDNIE SŁOWO W KAŻDEJ SYTUACJI

W katalogach wysyłkowych domów handlowych na Zachodzie reklamuje się to urządzenie jako elegancki upominek na każdą okazję. Przy zamówieniu 12 sztuk otrzymamy je (za dodatkową opłatą 1$) z wydrukowanym znakiem naszej firmy.

Elektroniczny słowniczek zawiera łącznie 7000 słów w pięciu językach: niemieckim, angielskim, francuskim, hiszpańskim i włoskim. Jest to w sumie po 1400 słów w każdym z tych języków.

Tłumaczenie odbywa się w bardzo prosty sposób. Najpierw należy po prostu napisać żądane słowo w wybranym języku, a potem naciskając kolorowe przyciski wskazać, na jaki język ma być przetłumaczone. 1400 słów to z pozoru niewiele, ale umożliwia całkiem swobodne poruszanie się za granicą przy wakacyjnych zakupach.



FOLIA OCHRONNA

Jak uchronić komputer przed skutkami przypadkowego wylania kawy lub herbaty na jego klawiaturę? Każdy właściciel komputera z przerażeniem myśli o niesłychanym spustoszeniu, jakie może spowodować nasz ulubiony napój po przedostaniu się pod obudowę. Jeśli spaliła się tylko jakaś kość pamięci, to możemy mówić o wyjątkowym szczęściu.

Czy wobec tego należy całkowicie zrezygnować z popijania herbatki podczas pracy lub zabawy z komputerem?

Okazuje się, że nie. Poradzono sobie z tym zagrożeniem w bardzo prosty sposób. Zachodnioniemiecka firma Planus oferuje swoim klientom specjalną folię ochronną „KeyScin". Folia ta chroni klawiaturę przed kurzem, rozlaniem wody, kawy, herbaty, popiołem z papierosów i wieloma innymi nieporządanymi substancjami. KeyScin umożliwia zupełnie normalną pracę z komputerem. Materiał, z którego wykonano folię, jest tak elastyczny, że naciśnięcie któregokolwiek klawisza nie powoduje przypadkowego uruchomienia sąsiednich klawiszy. Na spodniej stronie KeyScin znajdują się punkty pokryte specjalnym klejem. Dzięki temu folia po przyklejeniu do powierzchni komputera nie będzie się zsuwać.

Producent opracował już tego typu folie do ponad 200 typów komputerów. Wykonanie specjalne jest możliwe w terminie czterech do sześciu tygodni od zamówienia. Cena wynosi od 48 do 78 DM w zależności od wykonania.



PODRĘCZNY BANK DANYCH

To interesujące urządzenie elektroniczne, którego wymiary odpowiadają wielkości karty kredytowej, zastępuje jednocześnie notes i kalkulator. W jego pamięci można zapisać telefony, adresy i nazwiska 150 znajomych. Dla każdego numeru telefonu i adresu przewidziano po 12 znaków.

Czterodziałaniowy kalkulator z ośmiopozycyjnym wyświetlaczem z pewnością nikomu dziś nie zaimponuje, ale w połączeniu z poręcznym, elektronicznym notesikiem stanowi on interesujący upominek. Jest to zabawka, którą oferuje się w dużych ilościach z myślą o wykorzystaniu jako firmowy prezent reklamowy.

W związku z tym dystrybutor proponuje po przystępnej cenie wykonanie eleganckiego nadruku na aksamitnym futerale kalkulatorka.

KIESZONKOWY IBM



Firma SHARP oferuje swoim klientom bardzo pożyteczne narzędzie: kieszonkowy komputer z pamięcią o pojemności 32 KB i wyświetlaczem ciekłokrystalicznym 96 × 64 punktów. SHARP IQ-7100M to prawdziwe biuro tłumaczeń władające aż 8 językami. Rewolucyjnym rozwiązaniem zastosowanym w IQ-7100M są wymienne karty IC z oprogramowaniem oraz możliwość przyłączenia do komputera klasy IBM PC. Jakie zalety daje współpraca z PC, tego nie trzeba tłumaczyć nawet laikom. Standardowe funkcje komputera to: kalendarz na lata 1901–2099, terminarz roczny, miesięczny, tygodniowy i dzienny z zapamiętywaniem umówionych spotkań i możliwością robienia różnych notatek.

Dodatkowe karty to:
- słownik angielski podający 500 000 synonimów dla prawie 42 000 słów;
- program ułatwiający projektowanie skomplikowanych przedsięwzięć organizacyjnych, sprawowanie kontroli nad ich realizacją;
- ośmiojęzykowy tłumacz z zaprogramowanymi 480 zwrotami i 780 słowami z 13 różnych dziedzin życia;
- dodatkowa karta pamięci 64 KB.

KIESZONKOWY TELEWIZOR

Znak firmowy CASIO widnieje nie tylko na znakomitych dobrze znanych w Polsce kalkulatorach. CASIO oferuje również kolorowe kieszonkowe telewizorki z ekranem ciekłokrystalicznym. Producent proponuje 2 wersje odbiornika: TV 410 z ekranem o przekątnej 5 cm, lub większy TV 1400 7 cm.

Oba telewizorki umożliwiają odbiór audycji w zakresie fal VHF i UHF, automatyczne wyszukiwanie stacji emitującej program, regulację jasności obrazu i siły głosu. Jeśli znudzi się nam trzymanie odbiornika w ręce, to możemy go postawić na specjalnym stojaczku. Telewizor Casio jest zasilany z baterii, a na życzenie można dokupić dodatkowo zasilacz sieciowy.





ELEKTRONICZNY TŁUMACZ

Kieszonkowy komputerek HEXAGLOT M3 opanował aż 6 języków: angielski, francuski, hiszpański, portugalski, włoski i niemiecki. Jest to oferta dla bizmesmenów i turystów, którzy poważnie myślą o robieniu już w 1992 roku interesów i podróżach po zjednoczonej Europie. W pamięci HEXAGLOT-a zapisano łącznie 40800 słów. Ale proste podawanie odpowiedników jakiegoś wyrazu w 5 innych językach to nie koniec jego możliwości. Potrafi on wesprzeć swojego właściciela znajomością 150 najczęściej stosowanych w życiu codziennym zwrotów. Jedną z sześciu dziedzin, której ma dotyczyć zwrot, wybieramy bezpośrednio za pomocą oznaczonych odpowiednimi obrazkami klawiszy. Większość zdań jest tak zaprogramowana, że można je uzupełnić innymi słowami, które zapisane są w pamięci słownika.

Jeśli nie znamy dokładnej pisowni słowa, to wystarczy podać tylko jego 3 pierwsze litery. Komputer sam je odnajdzie i poda, jak się je pisze. Jeśli istnieje kilka słów zaczynających się od tych samych liter, to zostaną one wyświetlone w kolejności alfabetycznej.

Kalkulator, uzupełniający i tak niemałe możliwości elektronicznego tłumacza, pozwala na zaprogramowanie kursów 6 walut. Można wykonywać na nim nie tylko 4 podstawowe działania matematyczne, ale również bardzo szybko przeliczyć kwoty pieniędzy z jednej waluty na inną.

Niewielkie wymiary 13,8 × 6,8 × × 2 cm sprawiają, że HEXAGLOT z łatwością mieści się w kieszeni marynarki.

LOCOSCRIPT — DLA PC —

Amstrad PCW, opisywany kiedyś, jako coś więcej niż maszyna do pisania, jest jedynym komputerem 8-bitowym reklamowanym jeszcze na łamach profesjonalnych czasopism komputerowych. Sprzedany w samej Anglii w ilości ponad 700 tys. sztuk cieszy się olbrzymią popularnością, dzięki niskiej cenie i dużej użyteczności. Nie bez znaczenia jest fakt, że oferowany zestaw zawiera także oprogramowanie do edycji tekstu wliczone w cenę sprzętu. Jako ciekawostkę można podać, że jeden z przedstawicieli rosyjskiej pierestrojki — Borys Jelcyn — napisał swoją autobiografię posługując się rosyjskojęzyczną wersją programu LocoScript na komputerze Amstrad PCW.

Nic dziwnego, że po takich sukcesach firma Locomotive Software zdecydowała się na wprowadzenie na rynek nowej wersji programu LocoScript przeznaczonej na komputery klasy IBM PC. Rozprowadzany od lutego 1990 roku w cenie 143 funtów zestaw cieszy się dużą popularnością. Oprócz swoich podstawowych zadań program umożliwia realizację niewielkich baz danych, sprawdzanie poprawności ortograficznej tekstu i automatyczne adresowanie listów (mailmerge). Dzięki łatwości, z jaką można wprowadzać do programu obcojęzyczne czcionki (także cyrylica i jidysz), jest on niezwykle użyteczny dla wielu użytkowników.



2 MB RAM DO PCW

Do tej pory zwykle tylko komputery 16-bitowe wyposażane były w tak dużą pamięć typu RAM. W lutowym numerze angielskiego czasopisma **„8000 Plus"**, poświęconego wyłącznie komputerom Amstrad PCW, przedstawiono rozszerzenie pamięci operacyjnej dla tych właśnie komputerów.

Produkt, pod nazwą RamPack, dołączany do PCW przez złącze krawędziowe, oferowany jest przez znaną Czytelnikom Bajtka angielską firmę SCA Systems Ltd (patrz numer 3,89). W wersji podstawowej opisywane rozszerzenie zawiera 512 KB pamięci typu RAM. Pamięć ta jest wykorzystana na powiększenie Ramdysku, dostępnego zarówno przez LocoScript, jak i przez CP/M. 880 KB na dysku M: pozwala znacznie efektywniej posługiwać się programami takimi jak LocoScript 2, Protext, Master File i MiniOffice, nie wspominając o zwykłej pracy w systemie CP/M Plus.

Pakiet skonstruowany według zaleceń firmy Locomotive Software, zgodny jest także z programem Flipper Plus, pozwalającym na jednoczesną pracę dwóch różnych aplikacji na komputerze, z czasem przełączenia między nimi nie dłuższym niż dwie sekundy. W pierwszym kwartale tego roku oczekiwane jest rozszerzenie 2MB. Jak na razie zniechęca trochę wysoka cena wersji podstawowej — 512 KB — wynosząca 119 funtów.



AMIGA IS THE FUTURE

Sukces komputera Amiga na światowym rynku da się porównać jedynie z sukcesem słynnego Commodore 64. W roku 1987 liczba sprzedanych Amig wywindowała Commodore na pierwsze miejsce wśród producentów komputerów domowych i zapewniła firmie opanowanie rynku w ponad 80%. Zważywszy że Commodore otoczony jest równie dynamicznymi konkurentami, jak Atari, Schneider czy Apple, sukces ten zasługuje na uznanie. A rok 1987, to zdaniem ekspertów dopiero początek kariery, jaką Amiga ma przed sobą.

Historia powstania tego szlagieru na ciasnym rynku komputerowym jest równie pasjonująca, jak same techniczne możliwości Amigi i sięga roku 1982, kiedy to trzech dynamicznych wynalazców i biznesmenów: Jay Miner (były projektant układów scalonych w firmie Atari), Dave Morse (menadżer ds. sprzedaży w koncernie zabawkarskim Tonka Toys), i R.J. Mical (programista) założyli firmę Amiga Computer Inc. W Stanach Zjednoczonych szalała wówczas moda na gry komputerowo-telewizyjne i w związku z tym trzech panów postanowiło zbudować, jak sami to określili, „najbardziej zabójcze pudło do gier elektronicznych" i wprowadzić swój produkt na rynek. Należy pamiętać, że przy wspólnocie celów każdy z trzech założycieli Amiga Computer był wybitnym indywidualistą i nie znosił jakiegokolwiek podporządkowywania się innym. To był zresztą najistotniejszy powód, dla którego wszyscy porzucili intratne posady i postanowili się usamodzielnić.

W rezultacie Jay Miner wraz z zespołem techników zaczął projektować model nowego komputera, zamiast banalnej maszyny do gier. Przewidywał przy tym, że moda na gry szybko przeminie. Ponad rok jego wspólnicy byli utrzymywani w niewiedzy, nad czym naprawdę pracuje Jay Miner. Nie wiadomo, czym zakończyłoby się zachowanie jednego z trzech współwłaścicieli firmy, gdyby nie gwałtowne załamanie na rynku gier telewizyjnych. Moda prawie z dnia na dzień przeminęła, kto nie zdążył się przerzucić na coś innego, rychło ogłaszał plajtę. W tej sytuacji prawdziwy i wielostronny komputer był jedyną szansą dla firmy. Do współpracy zaangażowano inne firmy i powierzono im wykonanie mniej ważnych urządzeń, jak joystick czy mysz. Pracownicy Amiga Computer Inc. mogli się w pełni skoncentrować na tworzeniu swego komputera.

Premiera niezbyt jeszcze doskonałego modelu nastąpiła w styczniu 1984 podczas targów Consumer Electronics Show, przy drzwiach zamkniętych. Mimo, że Amiga pracowała jeszcze zbyt wolno, że cała elektronika zamontowana była na zwyczajnej platynie, że pudło komputera otoczone było prowizorycznymi antystatycznymi ścianami, fachowcy przyjęli go entuzjastycznie.

Niestety wszystko wskazywało na to, że sukces przyszedł odrobinę za późno: Amiga Computer Inc. stała w obliczu bankructwa. Intensywne badania spowodowały zadłużenie w wysokości 11 milionów dolarów. Jedynym wyjściem, poza ogłoszeniem plajty, była sprzedaż firmy. Do negocjacji zgłosili się przedstawiciele Sony, Apple, Philipsa, Silicon Graphics i Searsa. Najatrakcyjniej wyglądała oferta firmy Atari, która chciała spłacić długi, wypłacić kierownictwu Amigi półmilionową gratyfikację i wykupić akcje za połowę ceny nominalnej! Pomysł taki nie odpowiadał naturalnie menadżerom Commodore. Zdecydowanie wkroczyli do akcji z propozycją wykupienia akcji po cenie wyższej niż nominalna. Ich determinacja zaowocowała już po kilku tygodniach: nowo powstała firma Commodore-Amiga Inc. z kapitałem 27 milionów dolarów pracowała pełną parą nad usprawnieniem istniejącego już komputera. Do programu włączeni zostali najlepsi specjaliści z Commodore. Opracowanie systemu operacyjnego Amiga DOS powierzono sprawdzonej firmie Metacomco. Nową wersję Basica o nazwie Amiga Basic opracowywała firma Microsoft.

O tempie prac najlepiej świadczy termin kolejnej premiery, tym razem nie przy drzwiach zamkniętych, lecz w pełnych światłach jupiterów, z kampanią promocyjną w najlepszym amerykańskim stylu. Nastąpiła ona już w marcu 1984, w nowojorskiej operze. Cała prasa, telewizja, wszyscy przybyli na imprezę fachowcy wpadli w euforię, dziennikarzom brakło słów zachwytu. Natychmiast po prezentacji największe firmy produkujące oprogramowanie przystąpiły do opracowywania software'u. Już po kilku miesiącach było wiadomo, że Amiga podbiła Amerykę.

W 1985 roku nowy komputer został zaprezentowany Europie. Miało to miejsce w Londynie, pod hasłem „Amiga is the Future" (Amiga to przyszłość). Po raz pierwszy pokazano program Deluxe Paint, który pozwolił przekonać się naocznie o niebywałych możliwościach graficznych Amigi.

W 1986 roku w gmachu frankfurckiej opery Amiga została pokazana Europie, wkrótce potem pierwsze egzemplarze Amigi 1000 pojawiły się na rynku. Twórcy wiedzieli jednak, że to dopiero początek. Rozpoczęto intensywne przygotowania dwóch kolejnych modeli. Pierwszym z nich była wersja 500, miała zastąpić legendarny C-64; drugim miał być komputer będący konkurencją dla standardu PC. W chwili obecnej wszystkie planowane modele znajdują się w handlu i wszystkie (poza pierwszym, oznaczonym numerem 1000) znajdują się na liście komputerowych bestsellerów. Tak Amiga 500 jak i 2000 w wersji podstawowej posiadają 512 KB pamięci, przy czym oba modele dysponują możliwością zwiększenia jej. Granice praktycznie nie istnieją. Ostatnio np. oferowany jest twardy dysk do Amigi 500 i 2000 o pojemności — bagatelka — 750 megabajtów. Amiga zresztą od początku była pomyślana jako urządzenie do dalszej modularnej rozbudowy, o czym świadczy nie tylko bogaty zestaw gniazd na tylnej ścianie, ale także ukryte gniazda na bocznej ścianie.

Cała masa firm wyspecjalizowała się już w produkcji takich urządzeń, jeszcze więcej firm proponuje oprogramowanie, począwszy od gier komputerowych, po programy graficzne, muzyczne (Amiga może pomóc komponować), edycyjne (również programy z serii Desk Top Publishing), kalkulacyjne. Dla Amigi powstały już wersje najbardziej znanych profesjonalnych programów, jak Lotus 1,2,3, Word Perfect czy Multiplan, co pozwala na zastosowanie tego komputera w biznesie. W Amigi wyposażone są studia telewizyjne używające tego komputera do animacji i digitalizowania obrazów z taśmy video. Amiga jest zarazem znakomitym komputerem dla początkujących: Amiga Basic jest językiem łatwiejszym od tradycyjnego, nie wymaga na przykład numeracji linii podczas pisania programów, pozwala na łatwe tworzenie tzw. Pull-Down Menu, wprowadzanie myszy do programu, poruszanie obiektami, tworzenie ruchomych ciągów tekstowych itp.

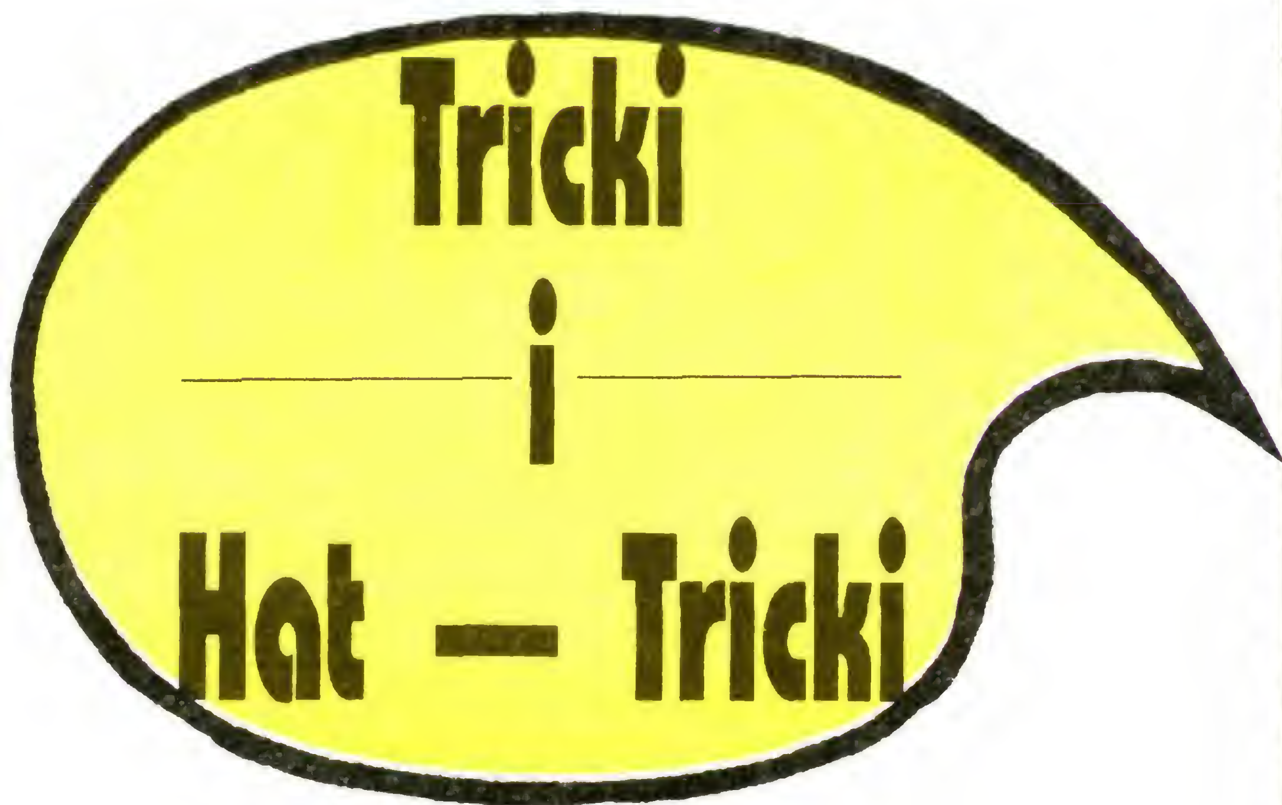
Do tego trzeba dodać, że już Amiga 500 (najmniejszy z proponowanych obecnie modeli) jest seryjnie wyposażona w 512 KB pamięci i stację dysków 3½" o pojemności 880 KB. Jej dodatkowymi plusami, nie spotykanymi dotychczas na rynku komputerów domowych są: tzw. multi-tasking function, co pozwala na jednoczesne rozwiązywanie kilku problemów; bezpośredni i łatwy dostęp do systemu operacyjnego (Amiga DOS) poprzez program zwany CLI (Command Line Interpreter); duża szybkość dokonywania skomplikowanych obliczeń graficznych oraz tzw. poziom intuicyjny, pozwalający na wykonywanie wielu programów poprzez piktogramy (icons). Chcąc, powiedzmy, w celu uzyskania dodatkowej pojemności, przechować duży tekst, wystarczy jego piktogram przesunąć za pomocą myszy na piktogram „kosza na śmieci" (Trashcan). Będzie on tam spoczywał do czasu opróżnienia kosza, na każde zadanie osiągalny. Chcąc otworzyć notatnik (Notepad — mini program edycyjny) wystarczy na jego piktogram naprowadzić strzałkę myszy i dwukrotnie nacisnąć klawisz.

W założeniu twórców Amiga miała być komputerem łączącym w sobie dwie, zdawałoby się przeciwstawne, cechy: profesjonalnej maszyny liczącej o dużej szybkości pracy oraz komputera domowego, zaspokajającego potrzeby natury zabawowej. To założenie sprawdziło się zaskakująco szybko. Amiga znalazła się w centrum zainteresowania przemysłu komputerowego, szybciej niż przypuszczano pojawiło się wiele rozwiązań, jak np. moduł gwarantujący kompatybilność ze standardem IBM PC.

Twórca Amigi, Jay Miner, pracuje nad dalszym udoskonalaniem swego komputera. Nowa wersja Amigi dysponuje pamięcią, mierzoną w wersji seryjnej w gigabajtach, ma także 16 megabajtów RAM, rozdzielczość 2048 x 2048 punktów (dziś Amiga osiąga „zaledwie" 620 x 400) oraz możliwość jednoczesnego odwzorowania na ekranie 65536 barw (z palety dysponującej ponad 16 milionami). Jay Miner przebąkuje coś również o wyposażeniu Amigi przyszłości w dysk optyczny o pojemności co najmniej 500 megabajtów.

Janusz Płoński





Wbrew Twoim przekonaniom stacja dysków do Commodore jest najbardziej zaskakującym urządzeniem wśród 8-bitowców. Praktycznie w każdym piśmie poświęconym sędziwemu już Commodore pełno jest różnych wspaniałych, ułatwiających obsługę sztuczek. W tym artykule postaram się przedstawić kilka najbardziej użytecznych i interesujących.

PROSTY LOADER

Każdy posiadacz stacji dysków dobrze wie, jakim wielkim wysiłkiem jest wgranie jakiegokolwiek programu, szczególnie jeśli nie wiadomo, co jest na danym dysku. Trzeba wgrać katalog, wylistować go, potem wytrwale „najechać" na tytuł kursorem, napisać przed nim LOAD, a za nim ,8 lub ,8,1. Zwykle zajmuje to mnóstwo czasu, siły, a co najgorsze — bardzo denerwuje.

Polecana przeze mnie metoda upraszcza te żmudne czynności. Podczas nagrywania czegokolwiek na dysk używaj podanych sekwencji:

SAUE"FILENAME‹SP› [D]8‹@›",8 — dodaje ,8
SAUE"FILENAME‹SP› [D]8[D]1‹@›",8 — dodaje ,8,1
‹SP› — Shift + Space
[D] — Commodore Key + D
‹@› — Shift + @

Jedynym ograniczeniem jest to, że nazwa pliku nie może mieć więcej niż 11 znaków.

SAVE + REPLACE

Wszyscy Commodorowcy znają ten problem — na dysku nie może być dwóch plików o tej samej nazwie, co niejednego przyprawiło już o dreszcze. Gdy więc na przykład piszesz swój program lub przepisujesz program z kartki do pamięci, bardzo często „sejwujesz" go, nie chcąc mieć kłopotów przy nagłym wyłączeniu prądu. Przy dość długich listingach tworzy się jednak problem, gdyż dysk jest tak zaśmiecony, że ciężko cokolwiek na nim znaleźć. Funkcja Scratch nie jest zbyt pomocna, ponieważ istnieje możliwość zawieszenia się komputera. Metodą na to może być:

SAUE"@:FILENAME",8

Aktualna zawartość pamięci zostanie wówczas umieszczona w pliku, którego nazwę podajesz w cudzysłowie (musi on jednak znajdować się na dysku!). To, co znajdowało się w nim przedtem, zostanie usunięte.

SELEKTYWNY KATALOG DYSKIETKI

Commodore DOS pozwala używać Ci specyficznych rozkazów, równie prosto jak normalnych opcji. Gdy więc nie interesuje Cię katalog całego dysku, a chcesz zobaczyć tylko określone pliki, jest to najzupełniej możliwe. Funkcje te stają się niezwykle przydatne, jeśli na dyskietce masz np. 100 rekordów, a chcesz obejrzeć wszystkie czteroliterowe. Oto kilka przykładów:

LOAD"$:NAME",8 — lista wszystkich plików o nazwie NAME
LOAD"$:CO*",8 — lista wszystkich nazw zaczynających się od liter CO
LOAD"$:*=S",8 — lista tzw. Sequential Files
LOAD"$:*=P",8 — lista tzw. Program Files
LOAD"$:MUSIC*=P",8 — pokazuje czołówki programów, zaczynających się od liter MUSIC
LOAD"$:??64",8 — pliki czteroliterowe, zakończone cyframi 64

DUBLOWANIE PROGRAMÓW

Często przydatną rzeczą może być przekopiowanie nagranego już pliku na ten sam dysk, inaczej go jednak nazywając. Jeśli masz np. plik o nazwie COOKS, który ciągle ulega zmianom, jest kasowany i nagrywany we wciąż innych wersjach, łatwo o jakąś pomyłkę czy przypadkowe zniszczenie. Aby temu zapobiec, możesz go zdublować i w odpowiednim momencie użyć. Używasz do tego następującej sekwencji:

OPEN1,8,15,"COPY:NEWFILE=OLDFILE": CLOSE1, podstawiając w cudzysłowie nazwy starego i nowego pliku.

ZABEZPIECZENIE

Jeśli posiadasz jakiś program i nie chcesz, aby niepożądane osoby wgrywały go podczas Twojej nieobecności, ten sposób zaspokoi Cię całkowicie. Należy napisać: OPEN5,8,5,"FILENAME,APPEND", wpisując w cudzysłowie, przed przecinkiem nazwę programu nagranego na dysku. Jeżeli będziesz chciał znowu z niego skorzystać, użyj następujących rozkazow: OPEN1,8,1,"FILENAME,P,R": POKE780,0: SYS65493: CLOSE1

DIRECTORY

Poniższy program jest bezcenny, w szczególności dla tych, którzy nie posiadają cartridge'ów lub innych rozszerzeń, ułatwiających dostęp do katalogu dyskietki. Nieszczęśliwcy Ci zmuszeni są do pisania niekończących się sekwencji LOAD"$",8 i następnie LIST, co przy n-tym dysku staje się nużące. Gdy więc leży przed tobą stos dyskietek do przejrzenia, skorzystaj z podanego programu.

```
10 OPEN1,8,0,"$"
20 GET#1,X$,X$
30 GET#1,X$,X$,X$,X$
40 IF ST THEN CLOSE1: END
50 GET#1,X$: IFX$="" THEN PRINT: GOTO30
60 IFX$=CHR$(34) THEN P=NOT P
70 IF P THEN PRINT X$;
80 GOTO50
```

"BLOCKS FREE"

Czasem, gdy przy użyciu Basic-a kopiujesz dużo krótkich plików, co pewien czas musisz sprawdzać ilość wolnego miejsca na dysku. Wymaga to kłopotliwego wgrywania katalogu, nierzadko nawet po kilka razy w ciągu minuty. Poniższy program znacznie ułatwia te czynności, po uruchomieniu wyświetla po prostu oczekiwaną przez Ciebie informację o ilości wolnych bloków.

```
10 INPUT"NAZWA PROGRAMU";N$
20 FORJ=1TO35: GET#10,X$: NEXT
30 GET#10,Y$:CLOSE10
40 BF =ASC(X$ + CHR$(0)) + 256*ASC(Y$ +
CHR$(0))
50 PRINT BF "BLOCKS FREE": END
```

ADRES STARTOWY

W pewnych sytuacjach niezbędne jest znalezienie adresu startowego programu i proponowany program robi to bardzo szybko i skutecznie.

```
10 INPUT"NAZWA PROGRAMU";N$
20 OPEN2,8,2,N$+",P,R"
30 GET#2,X$,Y$
40 PRINT ASC(X$ + CHR$(0)) + 256*ASC(Y$ +
CHR$(0))
50 CLOSE2
```

SENSOR URZĄDZENIA

Ten trick nie jest zbyt przydatny w praktyce, ale może zaspokoić wielu ciekawych, pragnących zgłębić wszystkie tajemnice komputera. Okazuje się bowiem, że napisanie PRINT PEEK(186) spowoduje wyświetlenie na monitorze numeru urządzenia, ostatnio przez nas używanego. Dzięki temu możesz dowiedzieć się na przykład, czy tajny program Twojego młodszego brata wgrany został z taśmy czy z dyskietki, etc.

OSTATNIA NAZWA

SYS62913 spowoduje wyświetlenie nazwy ostatnio wgranego przez Ciebie programu. Informacja ta, podobnie jak poprzednia jest mało użyteczna, ale mimo to warta odnotowania.

To już wszystkie ciekawostki przygotowane na dzisiaj. Postaramy się, aby już wkrótce znalazła się w „Bajtku" kolejna porcja ułatwień dla Commodore i nie tylko.

Na podstawie „Your Commodore"

Łukasz Czekajewski

TOMS MULTI DRIVE

Opisywana w „Bajtku" 1/90 stacja LDW Super 2000 została obecnie przez Panów Tadeusza Okonia (Warszawa, ul. Lachmana 4/44, tel. 641-54-29) i Marka Stolarskiego (Warszawa, ul. Kazury 13/26, tel. 46-01-02) wyposażona w najnowszą wersję systemu TOMS MULTI DRIVE.

System ten jest przeznaczony dla wszystkich stacji dysków Atari, które były lub są sprzedawane w Pewexie: Atari 1050, LDW Super 2000 i California Access 2001. **TOMS MULTI DRIVE** posiada wszystkie cechy **TOMS TURBO DRIVE**, a ponadto został wzbogacony o wiele nowych możliwości.

Przede wszystkim **MULTI DRIVE** pozwala na szybki zapis i odczyt dyskietek zapisanych standardowo, przez TURBO I MULTI, niezależnie od sposobu uruchomienia (zarówno przy zamkniętej, jak i przy otwartej stacji). Uzyskano to przez wyposażenie stacji we własną pamięć RAM o pojemności 8 KB, w której zapamiętywana jest zawartość całej ścieżki. W ten sposób odczyt ścieżki jest wykonywany podczas jednego obrotu dyskietki (standardowo potrzeba 9 do 18 obrotów). Dodatkowa pamięć RAM umożliwia ponadto zapamiętywanie historii pracy stacji, czyli wszystkich poleceń otrzymanych do wykonania od komputera.

Jeżeli uruchomimy komputer, gdy stacja posiadająca system **TOMS MULTI DRIVE** jest otwarta, to najpierw odczytywany jest ze stacji DOS, który znajduje się w jej pamięci ROM. Początkowo był to SuperDOS, a najnowsze wersje mają MYDOS 4.50. Ta część pamięci stacji działa jak dyskietka i można nawet odczytać jej katalog. Znajdują się tam ponadto dodatkowe programy pomocnicze i kopiujące. Oczywiście zapis do tej pamięci jest niemożliwy.

Niemal wszystkie nowe funkcje stacji można włączać i wyłączać przez naciskanie odpowiednich kombinacji klawiszy stacji. Szczegółowy opis funkcji tych klawiszy jest zawarty w instrukcji, która jest dołączana do przerobionej stacji.

OPROGRAMOWANIE

Znacznie wzbogacone zostało oprogramowanie dołączane do systemu **TOMS**, składa się na nie obecnie szesnaście programów. Są to przede wszystkim programy kopiujące i inicjujące. Wskażę tu tylko programy nie występujące w wersji **TURBO**.

TRACK MASTER służy do tworzenia zabezpieczeń przed kopiowaniem. Umożliwia on zapisanie sektorów zawierających błędy CRC, DDM lub RFN w dowolnej gęstości.

TRACER odczytuje w pamięci RAM stacji historię jej pracy. Jest ona zapamiętywana od momentu uruchomienia trybu śledzenia i pozwala na zapamiętanie około 3000 operacji.

INDEX FORMATTER pozwala na sformatowanie dyskietki przy wykorzystaniu otworu indeksowego. Umożliwia to kopiowanie programów zabezpieczonych metodą synchronicznego formatu.

GET TRACK I INDEX GET TRACK służą do odczytu w całości poszczególnych ścieżek dyskietki. Odczytywane są tu nie tylko bajty danych, lecz również wszystkie bajty pomocnicze.

PUT TRACK umożliwia z kolei zapisanie na dyskietce ścieżki ułożonej przez operatora. Zapis w tym wypadku dotyczy także całej ścieżki (wraz z informacjami pomocniczymi).

BAD FORMATTER zapisuje adresy sektorów na niesformatowanych ścieżkach dyskietki. Dzięki temu można szybko utworzyć dużą liczbę błędnych sektorów lub przygotować dyskietkę dla programu **WEAK WRITER**.

WEAK WRITER zapisuje częściowo sektory na dyskietce. Oznacza to, że początek sektora jest zapisany normalnie, a pozostały fragment jest niesformatowany. Zabezpieczona w ten sposób dyskietka jest niemożliwa do skopiowania i bardzo trudno ją odbezpieczyć.

HEX EDITOR służy do redagowania treści ścieżek odczytywanych przez **GET TRACK** i zapisywanych przez **PUT TRACK**. Umożliwia on między innymi na zmianę długości ścieżki.

KONWER jest programem dokonującym tłumaczenia plików tekstowych z Atari na IBM i odwrotnie. Sposób konwersji jest ustalany przez użytkownika. Program ten nie działa na Atari i musi być uruchamiany na IBM.

PODSUMOWANIE

Stacja z systemem **TOMS MULTI DRIVE** posiada możliwości, które w pełni może wykorzystać jedynie bardzo doświadczony programista. Przy jej pomocy można stworzyć zabezpieczenia dyskietki, które ze względu na nakład pracy niezbędny do skopiowania będą stanowiły skuteczną barierę dla piratów. Mniej zaawansowanych użytkowników z pewnością ucieszy większa szybkość pracy i jej komfort, brak kłopotów charakterystycznych dla LDW 2000 i możliwość współpracy z IBM.

Marek Zachar

Producenci:
Tadeusz Okoń, ul. Lachmana 4/44, Warszawa, tel. 641-54-29, Marek Stolarski, ul. Kazury 13/26, Warszawa, tel. 46-01-02.

Graficzny listing

Przedstawiony program umożliwia wydruk wszystkich znaków dostępnych na Atari w postaci, w jakiej są zapisane w generatorze znaków, na dowolnych drukarkach kompatybilnych z Epson FX-80 (np. Star NX, Star LC, Citizen, Seikosha SP180, Seikosha SP1200, Centronics GLP). Pozwala to drukować m.in. listingi wszelkich programów w Basicu, korzystać z dowolnego zestawu znaków itp.

Program tworzy w tabeli manipulatorów urządzenie o nazwie „H:". Możliwa jest zmiana maksymalnej długości wiersza (POKE 208,n; normalnie 80 znaków) oraz szerokości znaku (POKE 209,m; normalnie 1). Ta ostatnia wartość jest określana według tabeli:

m	gęstość druku	wysokość/szerokość
0	60 pkt/cal	1/1,2
1	120 pkt/cal	1/0,6
2	120 pkt/cal	1/0,6
3	240 pkt/cal	1/0,3
4	80 pkt/cal	1/0,9
5	72 pkt/cal	1/1,0
6	90 pkt/cal	1/0,8
7	144 pkt/cal	1/0,5

Cały program został podzielony na cztery części, które należy wykorzystywać zależnie od potrzeb. Listing 1 zawiera procedurę w języku maszynowym zapisaną w instrukcjach DATA i musi być użyty w każdym przypadku. Po połączeniu listingów 1 i 2 uzyskujemy program w Basicu do bezpośredniego wykorzystania.

Listingi 1 i 3 składają się na program, który zapisuje na kasecie wersję automatyczną. Po wczytaniu i zainstalowaniu urządzenia „H:" wersja ta uruchamia ponownie odczyt z magnetofonu, może więc być wykorzystana jako loader.

Po połączeniu listingów 1 i 4 powstaje program zapisujący procedurę obsługi w dyskowym pliku binarnym. Plik ten można uruchamiać funkcją „L" DOS-u lub dołączyć do innego programu w języku maszynowym funkcją Copy/Append (najlepiej na początku).

Ponadto listing 5 przedstawia wersję źródłową procedury w asemblerze ATMAS II. Dzięki temu można ją dołączyć do własnego programu w asemblerze lub wykonać wersję umieszczoną w innym miejscu pamięci.

UWAGA: Naciśnięcie klawisza RESET przerywa działanie procedury. Należy ją uruchomić ponownie instrukcją A=USR(1536) z Basica lub rozkazem JSR $0601 asemblera.

Jacek Żuk
Tomasz Żuk

listing 1

```
QC   101 DATA 104,169,1,133,209,169,80,133,
     998
MU   102 DATA 208,162,72,160,18,169,6,76,87
     1
ZY   103 DATA 188,238,33,6,6,255,192,254,11
     72
ZF   104 DATA 47,6,162,254,192,254,76,153,1
     144
RJ   105 DATA 254,0,165,209,41,7,141,203,10
     20
CQ   106 DATA 6,165,208,133,207,76,194,254,
     1243
CN   107 DATA 134,255,201,155,208,9,165,208
     ,1335
GX   108 DATA 133,207,169,155,76,203,254,19
     8,1395
QK   109 DATA 207,208,11,72,169,155,32,203,
     1057
RK   110 DATA 254,165,208,133,207,104,72,41
     ,1184
PD   111 DATA 127,32,110,6,24,162,0,134,595
YC   112 DATA 254,10,38,254,10,38,254,10,86
     8
QZ   113 DATA 38,254,133,253,173,244,2,101,
     1198
JN   114 DATA 254,133,254,76,127,6,201,32,1
     083
AZ   115 DATA 48,5,201,96,48,5,96,24,523
KB   116 DATA 105,64,96,56,233,32,96,162,84
     4
HC   117 DATA 7,169,0,157,206,6,160,7,712
CH   118 DATA 177,253,61,180,238,240,9,185,
     1343
MD   119 DATA 180,238,125,206,6,157,206,6,1
     124
SR   120 DATA 136,16,237,202,16,227,104,16,
     954
FC   121 DATA 13,162,7,189,206,6,73,255,911
WH   122 DATA 157,206,6,202,16,245,162,0,99
     4
SN   123 DATA 189,201,6,134,252,166,255,32,
     1235
TS   124 DATA 203,254,166,252,232,224,12,20
     8,1551
BP   125 DATA 239,189,201,6,166,255,76,203,
     1335
EQ   126 DATA 254,27,42,0,8,0,0,0,331
```

listing 2

```
AY   1 REM LISTING GRAFICZNY - BASIC
GJ   2 REM TOMASZ ZUK, JACEK ZUK
XO   3 REM COPYRIGHT (C) BAJTEK
NJ   4 REM
PF   5 M=1536:FOR I=1 TO 26
LP   10 A=0:FOR J=1 TO 8
IN   20 READ N:A=A+N:POKE M,N:M=M+1:NEXT J
RB   30 READ N:IF A<>N THEN ? :? "BLAD W LI
     NII ";100+I:LIST 100+I:END
PW   40 NEXT I:A=USR(1536):NEW
```

listing 3

```
ED   1 REM LISTING GRAFICZNY - CASSETTE
GJ   2 REM TOMASZ ZUK, JACEK ZUK
XO   3 REM COPYRIGHT (C) BAJTEK
NJ   4 REM
FK   5 DIM A$(256):M=1:FOR I=0 TO 28
LP   10 A=0:FOR J=1 TO 8
PA   20 READ N:A=A+N:A$(M,M)=CHR$(N):M=M+1:
     NEXT J
NQ   30 READ N:IF A<>N THEN ? :? "BLAD W LI
     NII ";98+I:LIST 98+I:END
ZF   40 NEXT I:OPEN #1,8,128,"C":? #1;A$;:C
     LOSE #1
KB   98 DATA 0,2,232,5,240,5,24,96,604
JQ   99 DATA 32,1,6,169,0,133,8,169,518
YB   100 DATA 1,141,233,3,76,110,198,0,762
```

listing 4

```
MB   1 REM LISTING GRAFICZNY - DISK
GJ   2 REM TOMASZ ZUK, JACEK ZUK
XO   3 REM COPYRIGHT (C) BAJTEK
NJ   4 REM
EF   5 DIM A$(256):M=1:FOR I=0 TO 27
MI   10 A=0:FOR J=1 TO 9
EA   20 READ N:IF N<256 THEN A=A+N:A$(M,M)=
     CHR$(N):M=M+1:NEXT J
VK   30 IF A<>N THEN ? :? "BLAD W LINII ";1
     00+I:LIST 100+I:END
SF   40 NEXT I:OPEN #1,8,0,"D:HDEVICE.OBJ":
     ? #1;A$;:CLOSE #1
VP   100 DATA 255,255,0,6,207,6,729
CS   127 DATA 226,2,227,2,1,6,464
```

listing 5

```
****************************
*                          *
* ATASCII-DRIVER           *
*                          *
* Assembler:ATMAS II       *
* Printer:SEIKOSHA SP-1200 *
*                          *
*              T.Zuk 1987 *
****************************

* Rejestry pomocnicze

COUNT     EQU $CF
LEN       EQU $D0
GRMODE    EQU $D1
MEX       EQU $FC
FIX       EQU $FD
KANAL     EQU $FF
CHBASE    EQU $02F4
MUSTER    EQU $EEB4

* Procedury OS

NEWDEV    EQU $EEBC
PHOPEN    EQU $FEC2
PHWRITE   EQU $FECB

****************************

          ORG $0600

* Ustawienie poczatkowych
* parametrow wydruku, nowy
* wpis do HATABS

          PLA
          LDA #$01      Tryb graficzny
          STA GRMODE
          LDA #$50      Dlugosc linii
          STA LEN
          LDX 'H        Nazwa urzadzeni
          LDY #PMAN     Adr.manipulator
          LDA #PMAN:H
          JMP NEWDEV

PMAN      DFW OPEN-1,$FF06,$FEC0,PUT-1
          DFW $FEA2,$FEC0,$994C,$00FE

****************************

OPEN      LDA GRMODE
          AND #$07
```

```
        STA LABEL+2
        LDA LEN
        STA COUNT
        JMP PHOPEN

* Kod znaku zapisany w aku-
* mulatorze, nr kanalu*16
* w rejestrze X

PUT     STX KANAL
        CMP #$9B       EOL?
        BNE A3
        LDA LEN
        STA COUNT
        LDA #$9B
        JMP PHWRITE
A3      DEC COUNT
        BNE A2
        PHA
        LDA #$9B
        JSR PHWRITE
        LDA LEN
        STA COUNT
        PLA
A2      PHA
        AND #$7F
        JSR ATASCR

* Obliczenie adresu przesylanego
* znaku w STANDCH. Wynik pod FIX

        CLC
        LDX #$00
        STX FIX+1
        ASL
        ROL FIX+1
        ASL
        ROL FIX+1
        ASL
        ROL FIX+1
        STA FIX
        LDA CHBASE
        ADC FIX+1
        STA FIX+1
        JMP ROTATE

* Zamiana kodu ATASCII na SCRCODE

ATASCR  CMP #$20
        BMI PLUS64
        CMP #$60
        BMI MIN32
        RTS
PLUS64  CLC
        ADC #$40
        RTS
MIN32   SEC
        SBC #$20
        RTS

* Obrot znaku

ROTATE  LDX #$07
LOOP    LDA #$00
        STA DEST,X
        LDY #$07
A4      LDA (FIX),Y
        AND MUSTER,X
        BEQ NEXT
        LDA MUSTER,Y
        ADC DEST,X
        STA DEST,X
NEXT    DEY
        BPL A4
        DEX
        BPL LOOP

* Inwersja znaku

        PLA
        BPL A6
        LDX #$07
A5      LDA DEST,X
        EOR #$FF
        STA DEST,X
        DEX
        BPL A5

* Transmisja znaku
* w trybie graficznym

A6      LDX #$00
A7      LDA LABEL,X
        STX MEX
        LDX KANAL
        JSR PHWRITE
        LDX MEX
        INX
        CPX #$0C
        BNE A7
        LDA LABEL,X
        LDX KANAL
        JMP PHWRITE

****************************

LABEL   DFB 27,42,0,8,0
DEST    ORG *+8
```

KILKA UWAG O Sparta DOS X

Mając już za sobą prawie rok praktyki w korzystaniu ze SpartaDOS X (nabyłem go w sierpniu 1989), pragnę podzielić się z Czytelnikami wnioskami i uwagami, jakie przez ten czas mi się nasunęły.

Osoby, które nie słyszały o **SpartaDOS X** informuję, że jest to dyskowy system operacyjny zapisany w module ROM (cartridge'u). Krótki opis jego możliwości był opublikowany w „Bajtku" 11/89.

Od pierwszej chwili zdumiała mnie — przede wszystkim — niesamowita wprost elastyczność tego systemu. Pozwala się on dopasować niemal do każdej konfiguracji sprzętu i do bardzo wielu programów. Oczywiście dobór odpowiedniego ustawienia parametrów jest trochę pracochłonny, po prostu trzeba próbować. Najgorzej wygląda w tym wypadku sprawa oprogramowania. Niestety, większość programów zabezpieczonych nie daje się przerobić na format **SpartaDOS**. Stosuję więc dwa rozwiązania: próbuję uruchamiać program z oryginalnej dyskietki za pomocą **Sparty**, a jeśli się nie uda, to zawsze pozostaje polecenie **COLD /N**, które ją po prostu odłącza. Niejako ubocznym efektem są prowadzone przeze mnie poszukiwania programów w wersjach niezabezpieczonych, które zwykle dają się uruchomić bez specjalnych zabiegów.

Można zapytać, czy gra jest warta świeczki. Oczywiście, że tak. Pokażę to na przykładzie najczęściej używanego przeze mnie programu — edytora **First XLEnt**. Przed wydrukowaniem napisanego tekstu trzeba „nauczyć" drukarkę polskich liter. W tym celu należy odczytać specjalny plik tekstowy i nakazać jego wydruk. Zawarte w nim kody sterujące, zawierają zestaw polskich liter dla drukarki. Czasem zdarzało mi się o tym zapomnieć i otrzymywałem tekst z dziurami, na przykład „pr dko " zamiast „prędkość". Teraz wygląda to zupełnie inaczej. Włączam komputer lub piszę **COLD** (kasuje to pamięć, jeśli używałem przedtem innego programu). **SpartaDOS** zakłada ramdysk numer 4 (XLEnt nie rozpoznaje większych numerów), pyta o datę i czas, a następnie głośno buczy (CONTROL+2) i nakazuje włączyć drukarkę. Teraz kopiuje polskie litery do drukarki i — na wszelki wypadek — do ramdysku oraz uruchamia edytor. Przedtem wszystkie te czynności musiałem wykonać ręcznie, teraz pamięta o nich komputer. Zakres czynności systemu można dodatkowo zwiększyć, np. instalując automatycznie procedury obsługi modemu, czy interfejsu RS232. Proszę pomyśleć, jaka rysuje się wspaniała perspektywa: napisany tekst (jeszcze ciepły) można wysłać z domu do redakcji nie zapisując go nawet na dyskietce.

Wspomniałem o wpisywaniu daty i czasu. Umieściłem takie polecenia na wszystkich dyskietkach z programami użytkowymi. Dzięki temu wiem dokładnie, co kiedy powstało i łatwo mogę znaleźć najnowszą wersję jakiegoś artykułu czy programu, a dodatkowo mam stale na ekranie zegar wskazujący aktualny czas. Ponadto archiwer wykorzystuje te informacje przy odświeżaniu zarchiwizowanych plików. Wadą jest konieczność wpisywania daty i czasu po każdym włączeniu komputera lub wyzerowaniu pamięci (COLD). Z tego właśnie powodu „choruję" teraz na moduł zegara **R-Time 8**, który robi to automatycznie.

Drugą „chorobą", którą wywołał u mnie **SpartaDOS X**, jest twardy dysk. Bez tego urządzenia system się po prostu marnuje. Z przeprowadzonych ze znajomymi elektronikami rozmów i obliczeń wynika, że napęd, kontroler i interfejs kosztowałyby w sumie około 400 dolarów, czyli tyle co dwie zwykłe stacje. Niestety, konieczne jest w tym celu zebranie co najmniej dziesięciu chętnych, gdyż przy wykonaniu pojedynczego egzemplarza koszty ogromnie wzrastają. Znalazłem już dwóch, więc jest nadzieja...

W zdobyciu pieniędzy na „twardziela" pomaga mi... sama **Sparta**. Jedną z jej największych zalet jest program archiwizujący. Dla pokazania korzyści posłużę się ponownie przykładem. Miałem dziewięć dyskietek z danymi do programów graficznych (zestawy znaków, rysunki i ikony) zapisanych dwustronnie w pojedynczej gęstości, gdyż niektóre programy innej gęstości nie rozpoznają. Po zarchiwizowaniu dane te zajmują półtorej dyskietki w podwójnej gęstości. W razie potrzeby odzyskanie takiego pliku trwa kilkadziesiąt sekund, łącznie z przepisaniem na „dyżurną" dyskietkę w pojedynczej gęstości. Gdy uda mi się namówić inne programy do pracy ze **SpartaDOS**, to zbędna będzie i ta dyskietka. Na razie „przekonałem" **Fun with Art, Design Master, PrintPower, B-Graph, SynGraph, SynStat, CalcMagic, HomeCard** oraz niemal wszystkie edytory tekstu i języki programowania. Przy okazji wyszło na jaw, że wiele programów zajmujących dwie strony dyskietki swobodnie mieści się na jednej — w podwójnej gęstości i bez DOS-u (bo zbędny).

Bardzo cenną zaletą jest możliwość dowolnego włączania i wyłączania **Basica** oraz innego modułu ROM. Posiadając **Action!** w ROM mam w każdej chwili dostęp nie tylko do doskonałego języka, lecz również do podręcznego edytora tekstów o całkiem niezłych możliwościach. Jest to wspaniałe ułatwienie przy konfigurowaniu systemu oraz przy pisaniu programów w różnych językach (programy w **Basicu** można także pisać w edytorze **Action!**). Dzięki zapisywaniu przez **SpartaDOS** w ramdysku całej zawartości pamięci wykorzystywanej przez te języki oraz strony zerowej i szóstej (!), nie trzeba się martwić, że zmienią się jakieś ustalone parametry (np. warianty pracy **Action!**), czy też dodatkowe procedury maszynowe z szóstej strony.

Oczywiście nie ma cudów i każdy produkt musi mieć jakieś wady. **SpartaDOS X** ma je również — nie można przekonfigurować systemu podczas pracy i brak pewnych poleceń (np. MOVE) albo niedoskonałe działanie innych (np. REMDIR — nie kasuje podkatalogów, które coś zawierają). Jest na to jednak sposób, ale pracochłonny. Należy napisać takie polecenia i podczas inicjowania systemu umieszczać je w ramdysku. Może to być nawet ramdysk specjalnie wydzielony (można założyć trzy jednocześnie). Następnie za pomocą PATH zmieniamy szlak poszukiwania poleceń tak, aby najpierw sprawdzany był ramdysk, a dopiero później moduł i nasze własne polecenia zastąpią oryginalne. Mam nadzieję, że po większym upowszechnieniu się **SpartaDOS** pojawią się również i takie rzeczy. Najbardziej brakuje mi jednoczesnego wyświetlania katalogów dwóch dyskietek, jak robi to **Norton Commander**. Wbudowane polecenie **MENU** wyświetla wprawdzie katalog i drzewo podkatalogów, ale tylko dla jednej stacji (jak **XTREE**).

Osobiście uważam, że jest to najlepsze urządzenie, jakie zakupiłem do swojego komputera od czasu nabycia stacji dysków. Natomiast do Czytelników posiadających **SpartaDOS X** zwracam się z prośbą o nadsyłanie uwag o pracy z tym systemem i propozycji usprawnień.

Wojciech Zientara

biblioteka Action! (5)

Na zakończenie cyklu o procedurach bibliotecznych Action! prezentujemy kilka procedur, które trudno zakwalifikować do jakiejś konkretnej grupy, niemniej jednak są one bardzo użyteczne. Są to:

Rand — generuje liczbę losową
Break — zatrzymuje program
Error — zastępuje instrukcję TRAP
Peek — odczytuje bajt z pamięci
PeekC — odczytuje słowo (2 bajty) z pamięci
Poke — zapisuje bajt w pamięci
PokeC — zapisuje słowo (2 bajty) w pamięci
Zero — zeruje blok pamięci
SetBlock — wypełnia wartością blok pamięci
MoveBlock — przemieszcza blok pamięci
device — określa ustalone urządzenie
TRACE — steruje opcją „TRACE" kompilatora
LIST — steruje opcją „LISt" kompilatora
EOF — określa stan wszystkich kanałów

Funkcja losowa

Generuje liczbę losową z podanego zakresu.
format: **BYTE FUNC Rand (BYTE zakres)**
parametry: **zakres** jest górną granicą generowanej liczby losowej.

Funkcja **Rand** zwraca przypadkową liczbę z zakresu od 0 do („zakres" – 1). Jeśli podany zostanie zakres 0, to zwracana jest liczba od 0 do 255.

Procedura Break

Zatrzymuje wykonywanie programu.
format: **PROC Break ()**
parametry: żadne

Procedura ta pozwala na zatrzymanie programu, na przykład w celu sprawdzenia wartości zmiennych lub skontrolowania poprawności jego realizacji. Wykonywanie zatrzymanego w ten sposób programu można wznowić przy użyciu polecenia **PROCEED** z monitora **Action!** Dalsza realizacja programu rozpoczyna się od instrukcji następującej po wywołaniu procedury **Break**.

Procedura Error

Procedura **Error** jest wywoływana automatycznie przez **Action!** po wystąpieniu błędu. Jeśli chcesz realizować obsługę błędów przez Twój program, musisz napisać procedurę i nakazać **Action!** stosowanie jej zamiast własnej procedury systemu. Wykonuje się to następująco:

```
PROC Trap(BYTE kod)
;to jest Twoja procedura obslugi bledow
;kod bledu, ktory wystapil, jest przez
;system Action! przekazywany jako jej
;parametr

;tu nastepuje obsluga bledu
;musisz to napisac samodzielnie
RETURN       ;koniec procedury Trap

PROC Program()
;glowna procedura Twojego programu

 CARD err ;przechowuje adres PROC Error

 err=Error  ;zapamietanie adresu
            ;procedury systemowej
 Error=Trap ;ustawienie adresu obslugi
            ;bledow na Twoja procedure

 ;tu znajduje sie Twoj program

 Error=err  ;odtworzenie adresu
            ;procedury systemowej
RETURN      ;koniec programu
```

W rzeczywistości wszystko, co robisz, to zmiana wskaźnika systemowej procedury obsługi błędów tak, aby wskazywał Twoją. Sam jednak nie możesz wywoływać tej procedury, gdyż jest ona wywoływana przez system **Action!** w wypadku wystąpienia błędu.

Warto zauważyć, że oryginalny adres procedury **Error** jest zapamiętywany i następnie — na końcu programu — ponownie odtwarzany. Trzeba tak zrobić, aby po zakończeniu programu powstałe błędy były obsługiwane przez procedurę systemową.

UWAGA: możliwość zastępowania procedury systemowej przez procedurę użytkownika jest bardzo cenna, lecz musi być, wykorzystywana bardzo ostrożnie. Nieuważne jej użycie może bowiem spowodować zawieszenie się systemu.

Funkcje Peek

Zwracają wartości odczytane ze wskazanej komórki pamięci.
formaty: **BYTE FUNC Peek (CARD adres)**
CARD FUNC PeekC (CARD adres)
parametry: **adres** jest adresem badanej komórki pamięci (lub pierwszej z dwóch badanych dla PeekC).

Funkcje te pozwalają na sprawdzanie zawartości pamięci podczas wykonywania programu. Pierwsza z nich odczytuje wartości jednobajtowe, druga zaś dwubajtowe (np. wektory lub adresy).

Procedury Poke

Zapisują podane wartości do wskazanych komórek pamięci.
formaty: **PROC Poke (CARD adres, BYTE liczba)**
PROC PokeC (CARD adres, liczba)
parametry: **adres** jest adresem zmienianej komórki pamięci (lub pierwszej z dwóch dla PokeC);
liczba jest wartością zapisywaną we wskazanej komórce.

Procedury te pozwalają na zmianę zawartości komórek pamięci podczas wykonywania programu. Pierwsza z nich zapisuje wartości jednobajtowe, druga zaś dwubajtowe (np. wektory lub adresy), przy czym młodszy bajt jest umieszczany w komórce „adres", a starszy w komórce „adres" + 1.

Procedura Zero

Wpisuje wartość zero do wskazanego obszaru pamięci.
format: **PROC Zero (BYTE POINTER adres, CARD rozmiar)**
parametry: **adres** jest wskaźnikiem początkowego adresu zerowanego bloku;
rozmiar jest długością zerowanego bloku.

Za pomocą tej procedury do wszystkich komórek pamięci wskazanego obszaru wpisuje się wartość zero. Obszar ten rozpoczyna się od komórki „adres" i kończy się na komórce „adres" + „rozmiar" – 1.

Procedura SetBlock

Wpisuje podaną wartość do wskazanego obszaru pamięci.
format: **PROC SetBlock (BYTE POINTER adres, CARD rozmiar, BYTE bajt)**
parametry: **adres** jest wskaźnikiem początkowego adresu zapisywanego bloku;
rozmiar jest długością zapisywanego bloku;
bajt jest wartością zapisywaną w bloku.

Za pomocą tej procedury do wszystkich komórek pamięci wskazanego obszaru wpisuje się podaną wartość „bajt". Obszar ten rozpoczyna się od komórki „adres" i kończy na komórce „adres" + „rozmiar" – 1.

Procedura MoveBlock

Przepisuje zawartość wskazanego obszaru pamięci do innego obszaru.
format: **PROC MoveBlock (BYTE POINTER cel, źródło, CARD rozmiar)**
parametry: **cel** jest wskaźnikiem początkowego adresu obszaru, do którego nastąpi przepisanie;
źródło jest wskaźnikiem początkowego adresu obszaru, z którego nastąpi przepisanie;
rozmiar jest długością przepisywanego bloku.

Procedura ta przepisuje zawartości wszystkich komórek pamięci obszaru rozpoczynającego się od „źródło" kończącego się na „źródło" + „rozmiar" – 1 do obszaru rozpoczynającego się od „cel" i kończącego się na „cel" + „rozmiar" – 1. Jeśli adres „źródło" jest większy niż „cel" i oba obszary częściowo się pokrywają, to procedura nie będzie działać poprawnie, gdyż część obszaru źródłowego zostanie wcześniej zapisana przez procedurę.

Zmienna device

device jest zmienną biblioteczną typu **BYTE**, która służy do określania ustalonego kanału **IOCB**. Wartość tej zmiennej określa numer kanału, przez który następuje wprowadzanie i wyprowadzanie informacji bez podawania numeru IOCB. Normalnie jest ona równa zero. Najłatwiej można wytłumaczyć jej działanie na przykładzie.

```
Open(5,8,0,"P:")
device=5
PrintE("to jest na drukarce")
device=0
PrintE("to jest na ekranie")
Close(5)
```



Obie procedury **PrintE** powodują zapisanie informacji na ustalonym urządzeniu. Podczas wykonywania pierwszej z nich jest to IOCB 5, a podczas drugiej ponownie IOCB 0. Mimo identycznej składni są więc one wykonywane na innych urządzeniach.

Zmienna TRACE

Ta zmienna biblioteczna jest także typu **BYTE** i służy do sterowania opcją „TRACE" kompilatora wewnątrz programu. Musi ona być użyta wraz z dyrektywą **SET** i musi znajdować się na początku programu. Przypisanie zmiennej **TRACE** wartości zero wyłącza śledzenie programu, a wartość jeden włącza je.
Przykład:

SET TRACE = 0

Zmienna LIST

Ta zmienna biblioteczna jest również typu **BYTE** i służy do sterowania opcją „LIST" kompilatora. Podobnie jak **TRACE**, musi ona być użyta wraz z dyrektywą **SET** i musi znajdować się na początku programu. Przypisanie zmiennej **LIST** wartości 0 wyłącza listowanie programu, a wartość 1 włącza je.
Przykład:

SET LIST = 1

Tablica EOF

W bibliotece **Action!** jest zdefiniowana jedna tablica ośmioelementowa:

BYTE ARRAY EOF (8)

Pozwala ona na stwierdzenie, czy w dowolnym kanale IOCB został już osiągnięty koniec pliku. Normalnie wszystkie elementy tablicy mają wartość zero. Po napotkaniu końca pliku w którymkolwiek kanale odpowiadający mu element tablicy otrzymuje wartość jeden. Dla jego zbadania wystarczy podać numer kanału jako indeks tablicy. Na przykład:

```
IF EOL(1) THEN
    Koniec()
  ELSE
    Dalej()
FI
```

powoduje wykonanie procedury Koniec (), jeśli cały plik otwarty w kanale 1 został już wykorzystany, w przeciwnym wypadku będzie wykonana procedura Dalej().

BIBLIOTEKA „BAJTKA"

Podczas pisania programów w **Action!** tworzy się wiele procedur, które mogą być stosowane również w innych programach. Zapisując te procedury można stworzyć sobie dodatkową bibliotekę procedur. Takie właśnie dodatkowe procedury **Action!** są od pewnego czasu publikowane w „Bajtku". Aktualnie biblioteka „Bajtka" zawiera następujące procedury:

obliczenia na liczbach rzeczywistych	11/88
procedury graficzne	12/88
uproszczone funkcje manipulatorów	12/88
funkcje **Abs** i **Sgn**	12/88
funkcje **Sin**, **Cos** i **Sqr**	1/89
procedury wejścia/wyjścia	3/89
procedury grafiki graczy i pocisków	7/89

Dalsze procedury tego rodzaju będą publikowane w następnych numerach „Bajtka", a w kolejnych wydaniach „Tylko o Atari" będziemy kontynuować kurs **Action!**

Wojciech Zientara

TAJEMNICE NIEŚMIERTELNYCH (4)

Po raz kolejny mam przyjemność prezentować poprawki od Michała Waydy z Krakowa:

SNOKIE
Ciąg liczb: A9, 03, 85, 8C zamieniamy na: A9, FF, 85, 8C (dziesiętnie: 169, 3, 133, 140 na: 169, 255, 133, 140) i zamiast trzech mamy 255 istnień.
SPINDIZZY
Aby zatrzymać upływ czasu, zamieniamy: A2, BC, 20, 65, 73 na: A2, BC, EA, EA, EA (dziesiętnie: 162, 188, 32, 101, 115 na: 162, 188, 234, 234, 234).
DRACONUS
Ciąg: E9, 01, 8D, 96, 3F zastępujemy: EA, EA, 8D, 96, 3F (dziesiętnie: 233, 1, 141, 150, 63 na 234, 234, 141, 150, 63) i mamy nieśmiertelność.
LODE RUNNER
Ciąg: CE, 92, 04 zamieniamy na AD, 92, 04 (dziesiętnie: 206, 146, 4 na: 173, 146, 4) i możemy popełnić dowolną ilość błędów. Uwaga! Poprawkę tę należy wprowadzić w trzech miejscach w programie.

Robert Wolnik ze Starego Bojarowa korzysta tylko z magnetofonu jako pamięci masowej — oto jego poprawki:
DEATH STAR
Nieśmiertelność w opcji dla jednego gracza uzyskamy przez zamianę: D6, 10 na B5, 10 (dziesiętnie: 214, 16 na: 181, 16).
W opcji dla dwóch graczy zamieniamy: D6, 89 na B5, 89 (dziesiętnie: 214, 137 na 181, 137).
GALACTIC CHASE
Ciąg: DE, C9, 20 zastępujemy: BD, C9, 20 (dziesiętnie: 222, 201, 32 na: 189, 201, 32).
EGGARD
Ciąg: C6, 8C zamieniamy na A5, 8C (dziesiętnie: 198, 140 na 165, 140)

ELEVATOR
CE, B9, 26 zastępujemy: AD, B9, 26 (dziesiętnie: 206, 185, 38 na: 173, 185, 38).

FIRE BUG
Sekwencję: C6, E3 zamieniamy na: A5, E3 (dziesiętnie: 198, 227 na 165, 227).
AIR SUPPORT
Ciąg: CE, 25, 84 zastępujemy: AD, 25, 84 (dziesiętnie: 206, 37, 132 na: 173, 37, 132).

Krzysztof Kobus ze Szczecina pomógł nam poprawić grę:
SEA DRAGON
Ciąg: D6, E0 zamieniamy: B5, E0 (dziesiętnie: 214, 224 na: 181, 224).
Albert Kucia ze Świdnicy poprawił:
SPELUNKER
C6, CB zastępujemy: EA, EA (dziesiętnie: 198, 203 na: 234, 234) i mamy nieskończoną ilość żyć.
ROBBO
Aby uzyskać 255 żyć, zamieniamy: A9, 08, 85, 9F na: A9, FF, 85, 9F (dziesiętnie: 169, 8, 133, 159 na: 169, 255, 133, 159).
Dodam od siebie, iż w tej grze można uzyskać 80 istnień bez ingerencji w program. Wystarczy przed uruchomieniem gry (w czasie wyświetlania planszy tytułowej z instrukcją) wpisać z klawiatury hasło — imię i nazwisko autora gry: JANUSZ PELC.

Z korespondencji Przemka Witkowskiego z Sosnowca dowiadujemy się, że w grze „Lode Runner" możemy sobie pomóc naciskając:

CTRL — F dodatkowe istnienie.
CTRL — U przejście do następnego poziomu.
CTRL — A wyratowanie się z pułapki (niestety ze stratą jednego istnienia)
CTRL — R zakończenie gry.

Tomasz Wiśniewski

U CHEMIKÓW

Z CHAOSU PORZĄDEK

Podglądaliśmy już astronomów i biologów korzystających z komputera. Tym razem zapuścimy sondę na twardym dysku u chemików. Czy komputery są im w ogóle do czegoś potrzebne? Oczywiście, i to jak! Współczesna chemia ma tyle wspólnego z chemią, którą znacie ze szkoły, co tabliczka mnożenia z matematyką. Chemicy w swojej pracy często rozwiązują skomplikowane równania różniczkowe, ale — zgodnie z obietnicą sprzed miesiąca — dzisiaj użyjemy komputera do czegoś innego. Posłużymy się modelem komórkowym i metodą Monte Carlo.

Zanim jednak sięgniemy po komputer, potrzebny będzie krótki wstęp teoretyczny. Od bardzo dawna (według archeologów od kilku tysięcy lat, choć może się to wydawać nieprawdopodobne) do pokrywania jednych metali drugimi stosuje się metody elektrochemiczne. Zasada jest bardzo prosta — do roztworu zawierającego jony metalu, którym chcemy pokrywać, wkłada się metalowy przedmiot, który chcemy pokryć, i dodatkową, obojętną elektrodę. Wystarczy teraz do obu elektrod przyłożyć odpowiednią różnicę potencjałów, żeby zaczął płynąć prąd, związany z redukowaniem (czyli — z naszego punktu widzenia — osadzaniem) jonów na pokrywanej powierzchni. Jony poruszają się w roztworze (dyfundują), a kiedy dotrą do powierzchni, ulegają na niej redukcji, czyli pobierają elektrony. (Oczywiście jest to pewne uproszczenie, muszą być spełnione różne dodatkowe warunki, żeby doświadczenie się powiodło, ale nie to nas interesuje.) Tym sposobem na jednym metalu otrzymujemy powłokę z drugiego metalu — na przykład cynkową na żelaznej rurze.

Takie metody stosuje się od bardzo dawna i od bardzo dawna wiadomo, że w zależności od tego, jaki metal pokrywamy jakim metalem, co znajduje się w roztworze i jaką przyłożymy różnicę potencjałów, powłoki będą wyglądały lepiej lub gorzej. Czasem będą lśniące i twarde, czasem matowe, a czasem będą wyglądać jak gąbka. Od czego to zależy? I co zrobić, żeby uzyskać ładną powłokę tam, gdzie wychodzi matowa?

Na takie pytania odpowiedzi można szukać na kilka sposobów. Jeden z nich to metoda prób i błędów, stosowana od setek lat przez rzemieślników. Ma ona oczywiste wady — tysiące prób na oślep mogą nie dać rezultatu, mimo że będziemy bardzo blisko sukcesu. Lepiej więc szukać rozwiązania w sposób systematyczny. Jedna z dwóch podstawowych możliwości to przeprowadzenie serii starannie zaplanowanych doświadczeń, na podstawie których można szukać odpowiedzi lub planować dalsze doświadczenia, zbliżające nas do rozwiązania. Druga możliwość to próba wytłumaczenia zjawiska na podstawie znanych faktów i teorii, a potem wykonanie doświadczeń potwierdzających (lub obalających) nasze przewidywania.

Spróbujmy zastosować tę drugą metodę w naszym przypadku. Stwórzmy model

tego, co się dzieje w roztworze, i sprawdźmy, czy uda nam się odtworzyć znane już zjawiska. Posłużmy się najprostszą możliwą symulacją obu zachodzących procesów — dyfuzji i redukcji.

Dyfuzja — to proces polegający na przenoszeniu się w roztworze (lub gazie) cząstek w taki sposób, który prowadzi do wyrównywania stężeń. Na ogół efekty zachodzenia dyfuzji, czyli zmiany stężenia w czasie, opisuje się za pomocą równań różniczkowych, ale w naszym modelu nie będziemy z nich korzystać. Zamiast opisywać skutki zjawiska, zajmiemy się jego przyczyną. A są nią chaotyczne ruchy i zderzenia wszystkich cząstek znajdujących się w roztworze. Ich chaotyczność powoduje, że nie potrafimy przewidzieć, gdzie za chwilę znajdzie się cząstka, którą obserwujemy. Jednak dzięki temu, że w roztworze takich cząstek są tryliony, możemy bardzo precyzyjnie opisać ich średnie zachowanie, właśnie za pomocą równań różniczkowych. Jeżeli będziemy symulować losowe ruchy wszystkich cząstek, z chaosu powinien wyłonić się porządek. Umówmy się więc, że naszym roztworem będzie ekran komputera. Każdy jego punkt to możliwe położenie jednej z cząstek, które nas interesują. Taka cząstka może losowo przenieść się do dowolnego z ośmiu sąsiednich punktów, może też zostać w miejscu. Tak skonstruowany model nazywa się modelem komórkowym — jednym z wielu możliwych. Gdyby któraś z cząstek usiłowała wydostać się poza ekran, to albo zostanie przez nas cofnięta z powrotem (co odpowiada sytuacji, w której odbiła się od ścianki naczynia), albo pojawi się z drugiej strony ekranu (co wprawdzie nie odpowiada żadnej konkretnej sytuacji, ale nie zmienia w niczym naszego modelu; możemy zresztą twierdzić, że jedna cząstka wyszła poza obserwowany obszar, a druga weszła do niego.)

Mamy już gotowy model dyfuzji (o niektórych uproszczeniach nie wspomniałem, ale mimo ich istnienia model działa prawidłowo). Została nam jeszcze redukcja cząstek na powierzchni elektrody. Zróbmy to tak — jeżeli na nowym miejscu (które zostałoby zajęte po wykonaniu chaotycznego ruchu) znajduje się już cząstka zredukowana, nasza cząstka dyfundująca przykleja się do niej, nie zmieniając miejsca, i od tego momentu jest też traktowana jak zredukowana. Jeżeli na ekranie będziemy zaznaczać tylko cząstki zredukowane, bardzo łatwo będzie napisać program realizujący taki model. Tylko czy na pewno nie popełniamy w naszym modelu żadnego błędu? Tak jak wygląda on teraz, każde uderzenie cząstki o powierzchnię elektrody powoduje jej redukcję, tymczasem w rzeczywistości tylko część zderzeń jest aktywna. Cząstka może uderzyć w elektrodę bardzo mocno i odbić się od niej nie ulegając reakcji, może się o elektrodę tylko otrzeć, nie stykając się z nią wystarczająco długo — trzeba to w jakiś sposób uwzględnić. Ponownie zrobimy to korzystając z chaosu — będziemy losować, czy cząstka ulegnie redukcji czy nie. Wprowadzimy w tym celu do modelu parametr, określający prawdopodobieństwo redukcji.

Teraz czas na przetłumaczenie naszego modelu na program. Ponieważ zależy nam na maksymalnej szybkości, dobrze będzie skorzystać z języka C, który jest niemal wymarzony do tego zadania. W dodatku, ponieważ nie są nam potrzebne inne liczby niż całkowite, w zupełności wystarczy kompilator HiSoft C na Spectrum (choć lepiej użyć szybszego komputera...). Spośród zdefiniowanych na początku stałych, najważniejsze są dwie — **prawd_red** i **maxred** określające prawdopodobieństwo redukcji cząstki uderzającej w elektrodę, i liczbę cząstek, po których zredukowaniu program ma przerwać działanie. Stała **PART** określa liczbę cząstek w roztworze (czyli pośrednio ich stężenie), pozostałe stałe opisują rozmiary ekranu w trybie graficznym. Z kolei wśród zmiennych najważniejsze są (poza tablicą **dane** [] zawierającą współrzędne cząstek i wskaźnikiem do jej elementów ***cząstka**) **nowex** i **nowey** opisujące nowe położenie cząstki po wykonaniu przez nią losowego ruchu. Zmienna **Nred** służy do liczenia zredukowanych cząstek, a **yred** (której istnienie nie wynika z modelu) pozwala na zaoszczędzenie czasu — dzięki niej tylko gdy cząstka znajduje się wystarczająco blisko elektrody, sprawdzane jest, czy w jej nowym położeniu znajduje się już jakaś zredukowana cząstka. Dzięki tej optymalizacji program jest kilka razy szybszy. Zastosowanie funkcji **Plot ()** i **Point ()** jest oczywiste. Funkcje **rnd ()** i **initrnd ()** to prosty generator liczb losowych, który opisałem osobno. Jego pseudolosowość przeszkadza podczas symulacji, więc żeby zniwelować jej wpływ, w każdym obiegu głównej pętli programu **(do...while)** wykonywany jest nowy posiew generatora.

Teraz trzeba uzbroić się w cierpliwość i uruchomić prezentowany program. Na Spectrum sto cząstek ulega redukcji w ciągu jednej godziny, toteż żeby móc obejrzeć ciekawe efekty swojej pracy, trzeba go włączyć co najmniej na dobę. Niecierpliwym radzę obejrzeć od razu rysunki, które w tym celu przygotowałem. Widać na nich, że im łatwiej cząstki wpadające na elektrodę ulegają redukcji, tym bardziej rozbudowane struktury (podobne do fraktali) powstają. Te struktury nazywają się dendrytami. Taka rozbudowana struktura to nic innego, jak gąbczasta powłoka. Gdy prawdopodobieństwo redukcji jest bardzo niskie, powstające powłoki są twarde i płaskie.

Tym sposobem udało się nam, nie wstając od klawiatury i korzystając z bardzo prostego modelu, w bardzo dosłowny sposób symulującego rzeczywistość, wskazać źródło kłopotów — jest nim zbyt wysokie prawdopodobieństwo redukcji cząstek na powierzchni elektrody. Teraz wypadałoby zająć się jego zmniejszaniem, ale to już zupełnie inna, nie komputerowa historia.

Marcin Borkowski

```
#define PART 1000
#define maxx 255
#define maxy 175
#define maxxp1 256
#define maxym1 174
#define prawd_red 50
#define maxred 1000

static char seed;
static reg1;

main()
{
   int i, j, nowex, nowey, Nred, yred;
   struct { char x, y; } *czastka, dane[PART];
   char rnd();

   inline(0xCD, 0xD6B);   /* cls */
   yred=1;  Nred=0;
   czastka = &dane[0];
   for (i=1; i!=41; ++i)
      for (j=1; j!=26; ++j)
      {
         czastka->x = 6*i;
         czastka->y = 7*j;
         ++czastka;
      }
   for (i=0; i!=256; ++i) Plot(i, 0);

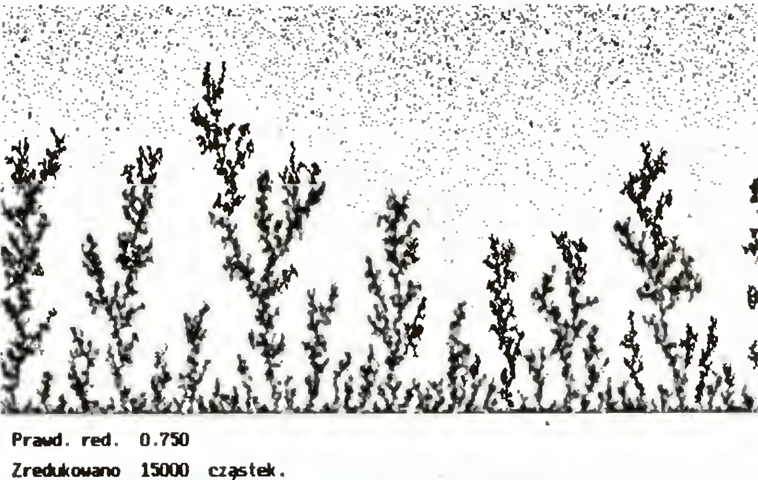
   do
   {
      czastka = &dane[0];
      initrnd();
      for (i=0; i!=PART; ++i)
      {
         nowex = (czastka->x + maxxp1 - rnd(3)) % maxx;
         nowey = czastka->y + 1 - rnd(3);
         if (nowey > maxy) nowey=maxym1;
         if ((nowey <= yred) && Point(nowex, nowey))
         {
            if (rnd(100) < prawd_red)
            {
               Plot(czastka->x, czastka->y);
               if (czastka->y > yred) yred = czastka->y;
               czastka->y = maxy;
               czastka->x = rnd(maxx);
               ++Nred;
            }
         }
         else
         {
            czastka->x = nowex;
            czastka->y = nowey;
         }
         ++czastka;
      }
   }  while (Nred < maxred);
}

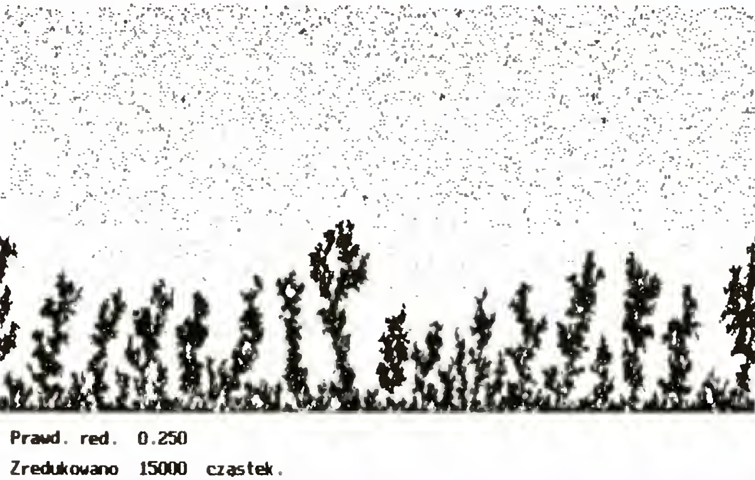
Plot(x, y)
{
   inline(0xDD, 0x46, 4, 0xDD, 0x4E, 6, 0xCD, 0x22E5);
}

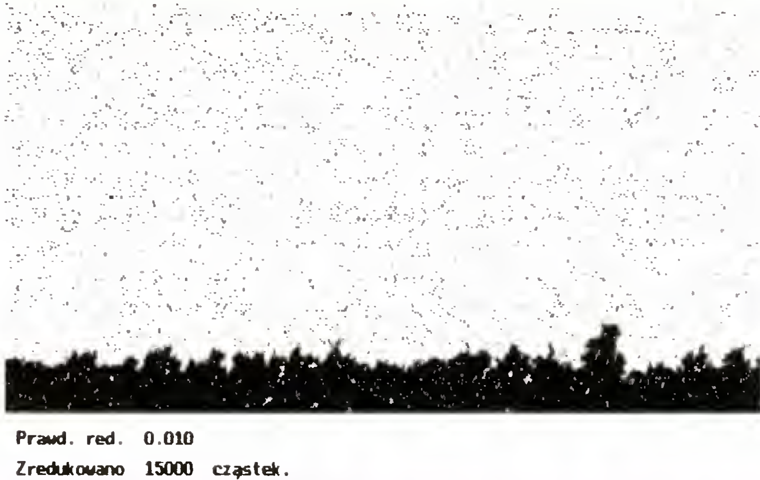
int Point(x, y)
{
   inline(0xDD, 0x46, 4, 0xDD, 0x4E, 6, 0xCD, 0x22AA,
          0x47, 4, 0x7E, 7, 0x10, 0xFD, 0x32, &reg1);
   return(reg1 & 1);
}

char rnd(ogr)
   char ogr;
{
   seed = 5 * ++seed;
   return(seed % ogr);
}

initrnd()
{
   inline(0xED, 0x5F, 0x32, &seed);
}
```









czyli szósty wieczór z czarnoksiężnikiem

Dziś nasz znajomy czarnoksiężnik miał bardzo dobry humor. Czytał właśnie bajkę o królewnie Śnieżce. Doczytał akurat do momentu, w którym uwielbiana przez niego czarownica otruła jabłkiem królewnę.

— Jeszcze kilka wieczorów, a będę umiał wywoływać duchy i będę mógł je poruszać. Wtedy rozprawię się z królewiczem. Smerfy też dostaną za znęcanie się nad Gargamelkiem.

Podczas ostatniego wieczoru nasz majster — czarnoksiężnik poznał instrukcję warunkową „gdy". Używając ją napisał dwa programy. Dziś zapragnął napisać program, który dokona „jakościowej" analizy wczytywanego tekstu. Analiza ta będzie polegała na obliczeniu, ile w danym tekście jest cyfr, „białych" znaków (spacji i tabulacji) i innych.

Pracę rozpoczął od obmyślania algorytmu.

— Najpierw muszę wczytać tekst. Następnie muszę wyzerować poszczególne liczniki. Główna część programu, w zależności od tego, jaki napotka znak, będzie zwiększała odpowiedni licznik. Czynność tę będzie powtarzała w pętli aż do końca tekstu.

W programie tym nie będzie żadnych nowych zaklęć, mogę więc przystąpić do pisania programu bez zmian w zbiorze <**predef.h**>. Oto ten program:

```
Program 1.

#include <stdio.h>
#include <predef.h>
majster
{ calkowite  cyfry,biale,inne,i;
  tekst t[80];
  czytaj ("%s",t);
  cyfry=biale=inne=0;
  powtarzaj (i=0; t[i]!='\n';i++)
    { gdy (t[i]>='0' && t[i]<='9') cyfry++;
      gdy (t[i]='\t' !! t[i]=' ') biale++;
       przeciwnie  inne++;}
  pisz ("\n cyfry %d, biale %d, inne %d", cyfry,biale,inne);
}
```

Program ten, mimo iż wykonuje swoje zadanie prawidłowo, ma pewną wadę. Otóż wszystkie warunki w instrukcjach „gdy" są sprawdzane zawsze, bez względu na fakt, czy wcześniejsze warunki były spełnione. Jeśli warunek z pierwszej instrukcji „gdy" został spełniony (tzn. rozpatrywany znak jest cyfrą), to pozostałe instrukcje można by pominąć i przejść do następnego cyklu pętli (czyli analizy kolejnego znaku).

Majster wziął księgę i zaczął szukać odpowiedniej instrukcji. W księdze było napisane: „W wypadku gdy chcemy powtórzyć kolejny cykl pętli z pominięciem dalszej jej części, musimy użyć instrukcji „cdn".

Odłożył księgę i przystąpił do modyfikacji programu. Najpierw jednak uzupełnił zbiór zaklęć. Do utworzonego wcześniej zbioru <**predef.h**> dopisał:

define cdn continue

Oto zmodyfikowany program:

```
Program 2.

#include <stdio.h>
#include <predef.h>
majster
{calkowite  cyfry,biale,inne,i;
 tekst   t[80];
 czytaj ("%s",t);
 cyfry=biale=inne=0;
 powtarzaj (i=0; t[i]!='\n';i++)
 { gdy (t[i]>='0' && t[i]<='9') cyfry++; cdn;
   gdy (t[i]='\t' !! t[i]=' ') biale++;
   przeciwnie inne++;}
 pisz ("\n cyfry %d, biale %d, inne %d",cyfry,biale,inne);
} i
```

Ten program wykona swoją pracę szybciej niż poprzedni. Majster nie był jednak z niego zadowolony. Mógł przecież utworzyć „drabinkę" złożoną z instrukcji „gdy". Dzięki takiej „drabince" mógł osiągnąć dokładnie ten sam efekt — po spełnieniu któregoś warunku, następne warunki nie byłyby już sprawdzane. Oto kolejna wersja tego programu:

```
Program 3.

#include <stdio.h>
#include <predef.h>
majster
{ calkowite  cyfry,biale,inne,i;
  tekst t[80];
  czytaj ("%s",t);
  cyfry=biale=inne=0;
  powtarzaj (i=0; t[i]!='\n';i++)
      { gdy (t[i]>='0' && t[i]<='9') cyfry++; przeciwnie
        gdy (t[i]='\t' !! t[i]=' ') biale++; przeciwnie
        inne++;}
      pisz ("\n cyfry %d, biale %d, inne %d", cyfry,biale,inne);
  }
```

Nasz majster miał awersję do drabin, zdarzyło się bowiem, że w dzieciństwie, gdy gonił kota, spadł z prawdziwej drabiny. Kot oczywiście uciekł.

Program z „drabinką" warunków eliminował dalsze sprawdzanie, nadal jednak sprawdzał wszystkie wcześniejsze.

— Musi być inny sposób — pomyślał czarnoksiężnik sięgając ponownie po księgę. Po niedługim szukaniu, z triumfem w oczach, odłożył księgę. Wyczytał w niej, że w wypadku rozpatrywania wyrażeń o wielu wartościach można przeskoczyć do wykonania odpowiedniego fragmentu programu dzięki instrukcji „wybierz". Ma ona postać:

wybierz (wyr)
{klucz w1: instr1; gotowe;
.
.
.
klucz wN: instrN; gotowe;
reszta: instrukcja;}

Instrukcja ta spowoduje, że program będzie wykonywał się od miejsca oznaczonego odpowiednim kluczem w1 ◊ wN, zgodnie z wartością wyrażenia wyr. Oczywiście, wykonałyby się także instrukcje oznakowane wszystkimi następnymi kluczami. Czasami tak chcemy, częściej jednak nie. Instrukcja „gotowe" kończy wykonywanie instrukcji „wybierz", pomijając pozostałe przypadki.

Wśród kluczy jest także klucz o nazwie „reszta". Oznacza on pozostałe, wcześniej niewyspecyfikowane wartości kluczy. Klucz ten można użyć w każdej instrukcji „wybierz" tylko jeden raz.

— Teraz muszę dopisać nowe zaklęcia — pomyślał czarnoksiężnik. Do zbioru <**predef.h**> dopisał:

#define wybierz switch
#define klucz case
#define gotowe break
#define reszta default

Wykorzystując te zaklęcia można napisać program, który będzie wykonywał tylko to, co trzeba i jednocześnie będzie czytelny, łatwy w zrozumieniu.

```
Program 4.

#include <stdio.h>
#include <predef.h>
majster
{calkowite  cyfry,biale,inne,i;
 tekst  t[80];
 czytaj ("%s",t);
 cyfry=biale=inne=0;
 powtarzaj (i=0; t[i]!='\n';i++)
   wybierz (t[i])
     gdy (t[i]='\t' !! t[i]=' ') biale++;
      { klucz '0':
        klucz '1':
        klucz '2':
        klucz '3':
        klucz '4':
        klucz '5':
        klucz '6':
        klucz '7':
        klucz '8':
        klucz '9': cyfry++; gotowe++;
        klucz '\t':
        klucz ' ': biale++; gotowe++;
        reszta: inne++;}
 pisz ("cyfry %d, biale %d, inne %d",cyfry,biale,inne);
} i
```

Widać, że użycie tej instrukcji może być kłopotliwe, gdyż zmusza do żmudnego wyliczania wszystkich możliwych przypadków. Są języki (np. PASCAL), w których wystarczy podać zakres wartości od — do. Nic nie stoi na przeszkodzie, żeby stworzyć kompilator języka „C", który także będzie akceptował taką lub podobną konstrukcję.

We wszystkich czterech programach użyto wielokrotnej instrukcji podstawienia:

cyfry = białe = inne = 0;

Język „C" jest tak miłym językiem, że przyjmuje, iż instrukcja podstawienia jest wyrażeniem, które ma wartość równą wartości podstawianej. Tak więc wspomnianą instrukcję można traktować jakby była napisana tak:

cyfry = (białe = (inne = 0));

W tym wypadku każde z wyrażeń w nawiasach równe jest zero. Instrukcję podstawienia możemy wykonać po wcześniejszym obliczeniu wartości wyrażenia z prawej strony znaku równości. Nic więc złego się nie stanie, jeśli pominiemy nawiasy.

Przy okazji warto wspomnieć o instrukcji skoku „teraz". Jest ona dyskwalifikowana przez propagatorów programowania systematycznego. Rzeczywiście — można pisać programy, które nie zawierają tej instrukcji. Zdarza się jednak, że osiąga się to kosztem bardzo rozbudowanych i często powtarzanych warunków. Nie stanie się tak, gdy użyjemy instrukcji skoku „teraz". Ma ona postać:

teraz ET;
.
.
.
ET: instrukcja;

Program wykonuje się instrukcja po instrukcji. Gdy dojdzie do instrukcji skoku, to teraz wykona instrukcję z etykietą ET i dalsze. Etykieta oddzielona jest od instrukcji dwukropkiem. Można kilka etykiet „przywiązać" do jednej instrukcji, np.

ET1:
ET2:
ET3: instrukcja;
lub **ET1: ET2: ET3: instrukcja;**

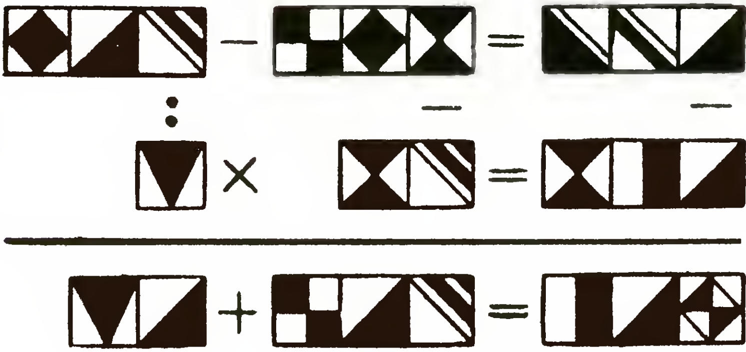
Do zbioru <**predef.h**> należy dopisać zaklęcie

#define teraz goto

Zauważyliście zapewne, że programy, które pisał dotąd czarnoksiężnik, rozwiązywały dość sztuczne problemy. Były one tylko ilustracjami zastosowań nowo poznanych zaklęć. Poza tym majster sam wykonywał cały program. Podczas następnego wieczoru zrzuci część pracy na swoich czeladników.

Mieczysław Płacheta

Algebraf — metody „czołgowe"



Rys. 1 Przykładowy Algebraf

Algebraf (rys. 1) jest zadaniem o charakterze matematyczno-logicznym, w którym symbole graficzne zastępują cyfry. Rozwiązanie polega na rozszyfrowaniu owych symboli i odtworzeniu wszystkich układów działań arytmetycznych. Należy pamiętać, że różnym elementom graficznym odpowiadają różne cyfry.

Zadanie to, zwykle wykonywane przez człowieka, w zależności od zdolności i od szczęścia, zajmuje od kilkunastu minut do paru godzin. Z matematycznego punktu widzenia problem znalezienia rozwiązania można sprowadzić do podania jednej z możliwych permutacji zbioru 10-elementowego. Łączna liczba tych permutacji wynosi 10!, czyli 3 628 800.

Dla człowieka jest to zadanie żmudne i nudne i w ten sposób raczej nie rozwiązywane. Jeśli jednak posłużymy się komputerem, to okaże się, że sprawa wygląda znacznie lepiej. Maszyna nie ma nic przeciwko bezmyślnemu powtarzaniu tych samych operacji i wykonuje je bardzo szybko. Prezentowany w artykule program znalazł rozwiązanie przykładowego algebrafu na komputerze IBM PC/XT w 20 sekund, sprawdzając ponad 77 tysięcy permutacji.

Od symboli graficznych do symboli matematycznych

Zastosowanie komputera wymaga, aby problem opisany za pomocą symboli graficznych i opisu słownego przetłumaczyć na język matematyki. Przedstawimy ogólny sposób postępowania na przykładzie konkretnego algebrafu.

Najpierw zastępujemy symbole graficzne pojedynczymi literami. Dla algebrafu z rys. 1 otrzymujemy:

```
ABC - DAE = FGB
 :     -     -
  H *  EC = EIB
-------------------
 HB + DBC = IBJ
```

Jeśli zastąpimy grupy liter zmiennymi, to nasz przykładowy algebraf będzie miał następującą postać:

```
M1  M2  M3
M4  M5  M6
M7  M8  M9
```

gdzie:

```
M1 = 100*A + 10*B + C
M2 = 100*D + 10*A + E
..........
M9 = 100*I + 10*B + J
```

W każdej grupie liter pierwsza od prawej reprezentuje cyfrę jedynek, druga od prawej cyfrę dziesiątek, a trzecia — cyfrę setek. Stąd biorą się właśnie mnożenia przez 10 i 100. W celu ujednolicenia zapisu wprowadźmy 10-elementowy wektor A, zamiast 10 zmiennych A, B, C,...J:

```
A -> A [1]
B -> A [2]
.......
J -> A [10]
```

Operacja ta pozwoli nam zapisać algebraf w następującej postaci:

```
M1 = 100*A[1] + 10*A[2] + A[3]
M2 = 100*A[4] + 10*A[1] + A[5]
..........
M9 = 100*A[9] + 10*A[2] + A[10]
```

Z kolei zależności arytmentyczne w algebrafie dadzą się przedstawić za pomocą 6 równań:

```
M1 - M2 = M3
M4 * M5 = M6
M7 + M8 = M9
M1 : M4 = M7
M2 - M5 = M8
M3 - M6 = M9
```

Równania te należy uporządkować tak, aby najpierw były działania prostsze: dodawanie i odejmowanie, a później mnożenie. Ewentualne dzielenie należy zastąpić mnożeniem. Takie postępowanie prowadzi do poniższych równań:

```
M1 - M2 = M3
M7 + M8 = M9
M2 - M5 = M8
M3 - M6 = M9
M4 * M5 = M6
M7 * M4 = M1
```

Permutacje

Przeprowadzone transformacje pozwoliły wyrazić wszystkie liczby w algebrafie za pomocą równań zawierających 10 elementów wektora A. Każdy taki element, zgodnie z warunkami zadania, może przyjmować jedną z 10 możliwych wartości: 0, 1,..9. Rozwiązaniem konkretnego algebrafu jest taka kombinacja (bez powtórzeń, czyli permutacja) elementów wektora A, dla której prawdziwe są wszystkie podane zależności arytmetyczne między liczbami.

Korzystając z komputera, będziemy szukać rozwiązania najprostszą (ale i najgłupszą) metodą, generując wszystkie możliwe permutacje 10-elementowego zbioru zawierającego 10 różnych cyfr. Dla każdej permutacji będziemy wyliczać liczby algebrafu i sprawdzać ich wzajemne zależności.

Jak generować permutacje?

Można to robić lepiej lub gorzej, łatwiej lub trudniej. Najprostsze rozwiązanie polega na wykorzystaniu n zagłębionych pętli FOR i sprawdzaniu, czy wszystkie elementy tak otrzymanej kombinacji są różne.

```
   FOR A=0 TO 9
     FOR B=0 TO 9
       IF B=A THEN GOTO XXXX  'SKOK DO NEXT B
       FOR C=0 TO 9
         IF C=A THEN GOTO XXX 'SKOK DO NEXT C
         IF C=B THEN GOTO XXX 'SKOK DO NEXT C
         '---------------------------------
         ' DALSZE PĘTLE
         ' DEFINICJA ALGEBRAFU
         ' SPRAWDZANIE WARUNKOW ALGEBRAFU
         '---------------------------------
   XXX NEXT C
XXXX NEXT B
   NEXT A
```

Niestety, przy takiej metodzie postępowania złożoność algorytmu wynosi 10^{10}, tzn. 10 miliardów. Jest to prawie trzy tysiące razy więcej niż 10!. Zakładając, że program generujący tylko permutacje wykonuje się w 1 godzinę, dochodzimy do wniosku, że program oparty na generowaniu kombinacji będzie działał prawie 3000 godzin, a to już jest raczej mało sensowne. Poszukajmy lepszego pomysłu.

Rekurencja

Dla wielu osób pojęcie rekurencji jest dość mgliste i raczej niezrozumiałe. Na szczęście łatwo wykazać błędność tego poglądu. Ogólnie biorąc, rekurencja jest procesem, w którym definiujemy pewną operację poprzez nią samą. Jako pierwszy przykład rozważmy pojęcie silni.

Zacznijmy od definicji iteracyjnej:

$N! = 1 * 2 * 3 \dots (N-1) * N$
$0! = 1$

Przykładowo:
$3! = 1 * 2 * 3 = 6,$
$4! = 1 * 2 * 3 * 4 = 24.$

Zauważmy, że $4! = 4 * 3!$. Widać, że możemy uogólnić ten wniosek i napisać, dodając warunek dla $0! = 1$

$N! = N * (N-1)!$
$0! = 1$

Taka definicja nosi nazwę rekurencyjnej, ponieważ zdefiniowaliśmy silnię przez silnię. To, co powoduje, że definicja taka nie jest absurdalna, wynika z faktu, że definiujemy silnię danej liczby jako iloczyn zawierający silnię z argumentem o jeden mniejszym i że mamy warunek stopu — 0! = 1.

Przykładowo:
$4! = 4 * 3!$
$3! = 3 * 2!$
$2! = 2 * 1!$
$1! = 1 * 0!$
$0! = 1$

Rekurencję można porównać do indukcji matematycznej. Metoda ta polega na dowodzeniu pewnych twierdzeń w następujący sposób:

— dowodzimy, że z prawdziwości twierdzenia dla n wynika prawdziwość twierdzenia dla n + 1,

— dowodzimy, że jest takie k, dla którego twierdzenie jest spełnione.

Jeśli potrafimy udowodnić te dwa warunki, to dowodzone twierdzenie jest prawdziwe, zaczynając od wartości k.

Można łatwo przekonać się, że generowanie permutacji też może być procesem rekurencyjnym. Po prostu permutacja k — elementowego zbioru jest równoważna wykonaniu k permutacji zbiorów o k−1 elementach. Sposób postępowania został zilustrowany dla k=4 na rysunku 2.

Zaczynamy od zbioru 4-elementowego (1, 2, 3, 4). Generujemy 4 grupy permutacji 3 elementowych. Każda z grup powstaje z zamiany k-tego (tu: czwartego) elementu z pozostałymi elementami. Podobnie generujemy permutacje 3-elementowe, jako grupę 3 permutacji 2-elementowych. Z permutacji 2-elementowej tworzymy dwie permutacje 1-elementowe. Ponieważ te ostatnie są po prostu tożsamościowe, proces ulega zakończeniu.

Program Permutation

Na listingu 1. przedstawiono program Permutation, generujący wszystkie permutacje zbioru n-elementowego. Najważniejszą jego częścią jest rekurencyjna procedura PERMUT. Jak widać, ta jednoparametrowa procedura woła samą siebie, ze zmniejszającym się argumentem. Dla k=1 proces ulega zatrzymaniu i wołana jest procedura PRINT, drukująca otrzymaną permutację. Warunki początkowe (pierwsza zadana permutacja) definiowane są w procedurze INIT, zadającej wektor A, reprezentujący elementy zbioru podlegającego permutowaniu.

Na rys. 3 przedstawiono przebieg testowy wykonany na komputerze IBM PC. Program napisano w

Turbo Pascalu i uruchomiono za pomocą kompilatora w wersji 3.0 na komputerach IBM PC i Amstrad PCW (CP/M 80). Szybkość działania na IBM-ie z zegarem 10 MHz wynosi około 120 000 permutacji na min, a na Amstradzie jest 10 razy mniejsza. Zastosowanie najnowszej wersji Turbo 5.5 przyśpiesza program o 50%. W wypadku Amstrada lub innego komputera CP/M-u należy kompilować program z opcją ($A–) (kod nie absolutny).

Program Algebraf

Zastępując procedurę Print procedurą EVAL, która oblicza elementy Algebrafu i sprawdza jego warunki, otrzymujemy program, dzięki któremu możliwe jest znalezienie rozwiązania przykładowego algebrafu. Procedura Permut została usprawniona poprzez wprowadzenie szybszej instrukcji MOVE, służącej do zapamiętania fragmentu wektora A w wektorze B. Manipulując procedurą INIT jesteśmy w stanie zmniejszyć czas działania programu, jeśli mamy dodatkowe informacje o rozwiązaniu. W najgorszym wypadku procedura Permut wykona się 3 628 800 razy, co na komputerze PC/XT nie zajmie więcej niż 20 minut (zegar 10 MHz, Turbo Pascal 5.5).

Rozwiązanie innego algebrafu wymaga zmiany wyłącznie w procedurze EVAL, w części definiującej liczby i warunki algebrafu. Program może służyć do znajdowania rozwiązań dowolnych algebrafów omawianego typu. Przedstawiony algorytm generacji permutacji trudno zrealizować w BASIC-u, w równie przejrzysty sposób, nie dysponując procedurami z parametrami i zmiennymi lokalnymi.

JONASZ MAYER

```
{1,2,3,4}

    |
    V

{1,2,3} {4}

{1,2,4} {3}  ==>  {1,2} {4} {3}
                  {1,4} {2} {3}
{1,4,3} {2}       {4,2} {1} {3}  ==> {4} {2} {1} {3}
                                     {2} {4} {1} {3}
{4,2,3} {1}
```

Rys. 2 Znajdowanie permutacji 4-elementowego zbioru

```
Compiling
  83 lines

Code:       0027 paragraphs (  624 bytes), 0D01 paragraphs free
Data:       0003 paragraphs (   48 bytes), 0FD9 paragraphs free
Stack/Heap: 7062 paragraphs (460320 bytes)

>

Running
Podaj liczbe permutowanych elementow: 3
      1   0 1 2
      2   1 0 2
      3   0 2 1
      4   2 0 1
      5   2 1 0
      6   1 2 0
Podaj liczbe permutowanych elementow: 2
      1   0 1
      2   1 0
Podaj liczbe permutowanych elementow: 1
      1   0
```

Rys. 3 Kompilacja i wykonanie programu Permutation

```
Compiling
  111 lines

Code:       0046 paragraphs ( 1120 bytes), 0CE2 paragraphs free
Data:       0004 paragraphs (   64 bytes), 0FD8 paragraphs free
Stack/Heap: 6051 paragraphs (394512 bytes)

>

Running
 9 8 7 6 5 4 3 2 1 0
10000 7 4 5 8 6 2 9 3 1 0
20000 2 3 6 7 4 8 9 5 1 0
30000 8 4 6 5 3 2 9 7 1 0
40000 4 3 5 8 6 7 2 9 1 0
50000 5 1 4 7 8 6 9 3 2 0
60000 8 7 6 4 3 1 9 5 2 0
70000 1 3 4 8 6 5 9 7 2 0

 837 184 653
   9  47 423
  93 137 230

>
```

Rys. 4 Kompilacja i wykonanie programu Algebraf

```
program Permutation;
{********************************************}
{                                            }
{      Program generujacy permutacje         }
{        k-elementowego wektora              }
{                                            }
{            (C) JM 1989                     }
{                                            }
{********************************************}
const
  Max = 10;        { maksymalna liczba elementow }
type
  vector = array [1..Max] of byte;
var
  A     : vector;
  k     : byte;
  count : integer;         { licznik permutacji }

procedure Init;
{********************************************}
{ Procedura inicjalizuje licznik permutacji i }
{ wektor elementow.                          }
{********************************************}
var i : byte;
begin
  count := 0;
  for i := 1 to 10 do A [i] := i-1;
end;  { of Init }

procedure Print;
{********************************************}
{ Drukuje wektor elementow.                  }
{********************************************}
var i : byte;
begin
  count := count + 1;
  write (count:7,'   ');
  for i := 1 to k do write (A[i]:2); writeln;
end; { of Print }

procedure Permut (k : byte);   {$A-} {CP/M 80}
{********************************************}
{ Procedura generuje w sposob rekurencyjny   }
{ wszystkie, mozliwe permutacje zbioru       }
{ k-elementowego. Permutacja zbioru 1-elemen- }
{ towego jest operacja tozsamosciowa. Permu- }
{ tacja zbioru k-elementowego jest realizowa- }
{ na jako k permutacji (k-1)-elementowego    }
{ zbioru                                     }
{********************************************}
var i,j, x : byte;
    B      : vector;
begin
  if k=1
  then Print
  else begin
       { zapamietanie k-elementowego wektora }
       for j := 1 to k do B [j] := A [j];

       for i := 1 to k   { kolejne permutacje }
       do begin

          { odtworzenie wektora }
          for j := 1 to k do A [j] := B[j];

          x := A [k+1-i];        {  zamiana  }
          A [k+1-i] := A [k];    { elementow }
          A [k] := x;            { wektora A }

          Permut (k-1);
        end;
     end;
end;  { Permut }

{********************************************}
begin   { main }
  repeat
    write ('Podaj liczbe permutowanych elementow: ');
    readln (k);
    Init;
    Permut (k);
  until k = 1;
end.
{********************************************}
```

Listing 1. Plik PERMUT.PAS

```
program Algebraf;
{********************************************}
{                                            }
{         Rozwiazywanie algebrafow           }
{                                            }
{            (C) JM 1989                     }
{                                            }
{********************************************}
const
  Max = 10;        { maksymalna liczba elementow }
type
  vector = array [1..Max] of byte;
var
  A : vector;
  count, count2 : integer;
  m1,m2,m3,m4,m5,m6,m7,m8,m9 : integer;

procedure Init;
{********************************************}
{ Procedura inicjalizuje licznik permutacji i }
{ wektor elementow.                          }
{********************************************}
var i : byte;
begin
  count := 0; count2 := 0;
  for i := 1 to 10 do A [i] := 10-i;
  for i := 1 to 10 do write (A[i]:2);   writeln;
end; { of Init }

procedure Results;
{********************************************}
{ Wydruk rozwiazania.                        }
{********************************************}
begin
  writeln;  writeln (m1:4,m2:4,m3:4);
            writeln (m4:4,m5:4,m6:4);
            writeln (m7:4,m8:4,m9:4);    halt;
end; { of Results }

procedure Eval;
{********************************************}
{ Zliczanie permutacji i sprawdzanie dla danej }
{ permutacji warunkow spelnianych przez algebraf }
{********************************************}
var i : byte;
begin
    { zliczanie permutacji }
  count := count + 1;
  if count=10000
  then begin

       count := 0; count2:=count2+1;
       write (#13,count2,'0000');
       for i := 1 to 10 do write (A[i]:2);
       writeln;
     end;

     { definicja algebrafu }

  m1 := 100*a[1] + 10*a[2] + a[3];
  m2 := 100*a[4] + 10*a[1] + a[5];
  m3 := 100*a[6] + 10*a[7] + a[2];
  m4 :=                      a[8];
  m5 :=            10*a[5] + a[3];
  m6 := 100*a[5] + 10*a[9] + a[2];
  m7 :=            10*a[8] + a[2];
  m8 := 100*a[4] + 10*a[2] + a[3];
  m9 := 100*a[9] + 10*a[2] + a[10];

  { warunki spelniane przez algebraf }

  if m1-m2=m3
  then if m7+m8=m9
       then if m2-m5=m8
            then if m3-m6=m9
                 then if m4*m5=m6
                      then if m4*m7=m1
                           then Results;
end;  { of EVAL }

procedure Permut (k : byte);   {$A-} {CP/M 80}
{********************************************}
{ Procedura generuje w sposob rekurencyjny   }
{ wszystkie, mozliwe permutacje zbioru       }
{ k-elementowego. Permutacja zbioru 1-elemen- }
{ towego jest operacja tozsamosciowa. Permu- }
{ tacja zbioru k-elementowego jest realizowa- }
{ na jako k permutacji (k-1)-elementowego    }
{ zbioru                                     }


{********************************************}
var i, x : byte;
    B    : vector;
begin
  if k=1
  then EVAL
  else begin
       Move (A,B,k);
       for i := 1 to k  { kolejne permutacje }
       do begin
          Move (B,A,k);
          x := A [k+1-i];        {  zamiana  }
          A [k+1-i] := A [k];    { elementow }
          A [k] := x;            { wektora A }
          Permut (k-1);
        end;
     end;
end;  { Permut }

begin { main }
   Init;
   Permut (10);
end.
{********************************************}
```

Listing 2. Plik ALGEBRAF.PAS



Yoy

WYNIKI TURNIEJU JOYSTICKÓW

Na ogłoszony w dziewiątym numerze „Bajtka" z 1989 r. Turniej Joysticków otrzymaliśmy trzy zgłoszenia.

Oznacza to dwie rzeczy. Po pierwsze, popyt na joysticki nie powoduje w Polsce zwiększenia produkcji, lecz zwiększenie importu. Po drugie, sprowadzanie jest w każdym razie tańsze i mniej kłopotliwe niż produkcja.

Szkoda więc, że nie otrzymaliśmy ani jednego zgłoszenia od importerów joysticków. Pozwala to stwierdzić, że import ten jest „dziki", a więc przypadkowy, nie opodatkowany, prowadzony przez pojedyncze osoby. Nie zapewnia więc obsługi gwarancyjnej, serwisu, informacji. Oczywiście zepsuty joystick wyrzuca się najczęściej do kosza, ale nie każdy może sobie na to pozwolić. Wyniki testu zachodnioniemieckich kolegów, publikowane w poprzednim numerze, dowodzą, że nie każdy joystick psuje się dopiero po roku intensywnego grania.

Napisały do nas trzy firmy:

— **ELEKTROMECHANIKA z Dobczyc, ul. Cegielniana 17,**
— **JOYSTICK SERVICE — Warszawa 130, skr. poczt. 102,**
— **MATT z Łodzi, ul. Wigury 15.**

Pierwsza i trzecia firma zajmuje się produkcją, druga zaś naprawą joysticków (głównie jest to montowanie mikroprzełączników po niewielkich cenach). Otrzymaliśmy do testowania po kilka egzemplarzy — przykładów oferowanej przez firmy usługi. Ponieważ w wypadku trzech zawodników nawet ostatni otrzyma medal, postanowiliśmy odejść od modelu wielostronnych, wykańczających testów. Otrzymane wyniki nie stanowiłyby podstawy do autorytatywnego porównania, a w konsekwencji do wyłonienia najlepszego produktu.

Nasz Turniej polegał więc na kilku normatywnych i kilku zwariowanych próbach. Zacznijmy od tej drugiej grupy:

1. Marznięcie. Joysticki wisiały 14 dni na mrozie −10°C. Natychmiast po zdjęciu działały bez zarzutu. Po dwóch dniach produkt firmy ELEKTROMECHANIKA, joystick JOY, miał kłopoty z przyciskiem FIRE. Pozostali zawodnicy zachowywali się standardowo.

2. Piekarnik. Wnętrze joysticka zostało doprowadzone do temperatury 70°C (mierzona termoparą). Natychmiast po wyjęciu kłopoty sprawiał jedynie MATT. Po pół godzinie wrócił do normy.

3. Woda. Godzinne moczenie w 10% roztworze NaCl. Bez kłopotu kontaktował JOY, MATT zaś i przerabiany przez Joystick Service egzemplarz kontaktowały po mocnym dociśnięciu. Następnego dnia nie kontaktował żaden, nawet MATT.

4. Kręcenie nad głową. Nie udało się zniszczyć żadnego joysticka. Prędkość kątowa 2 obr/sek przy trzymaniu za wtyczkę nie powoduje żadnych skutków.

5. Dzieci. Trzej najwięksi mali kaci joystików otrzymali trzy egzemplarze. Po tygodniu ferii zimowych nie działał żaden. W przypadku JOY-a było to zerwanie kabli, MATT-a — wyrwanie rączki, produktów Joystick Service — totalna masakra.

Jeśli chodzi o rzeczywiste próby użytkowe, to przeprowadziliśmy:

1. 10.000 szybkich wciśnięć FIRE. Wszystkie trzy przeżyły.

2. Test szczelności przyssawek. W przypadku Joystick Service przyssawki zależą od producenta, więc test nie ma znaczenia. Przyssawki JOY-a są bardzo słabe, na poziomym stole trzymają 2 sekundy, na szybie zaś ani chwili. Natomiast MATT jest wyposażony w przyssawki dobrej jakości — na stole 1.5 minuty, na szybie 28 sekund. Siła potrzebna do oderwania joysticka od stołu wynosi średnio 30 N, tj. ponad dwa razy więcej, niż potrzeba do pewnego zadziałania styku.

3. Ocena długości kabla. W wypadku produktów Joystick Service nie ma to również znaczenia. JOY posiada krótki, gruby, nieelastyczny kabel, niewygodny w akcji. Zrzucenie JOY-a ze stołu powoduje znaczne przesunięcie komputera, możliwe jest też wyrwanie przewodu z gniazda lub interface-u (Spectrum). Co do MATT-a, to kabel jest długi i cienki, co pozwala na wygodną manipulację i upadki.

4. Test czułości. Najlepiej sprawują się joysticki z mikroprzełącznikami montowanymi przez Joystick Service. Drugie miejsce ex aequo przyznaliśmy MATT-owi i JOY-owi. Pierwszy posiada znaczny kąt wychyłu, ale do zwarcia styku potrzebna jest spora siła. Drugi zaś działa właściwie po lekkim dotknięciu, lecz kąt pracy jest bardzo niewielki. Przy tym kłopoty są z kierunkami pośrednimi.

5. Ocena wygody użytkowania. MATT jest bardzo wygodny do trzymania w dłoniach lub opierania o stół. Kształt jego jest obły i fizjologiczny, przez co nie powoduje szybkiego zmęczenia. Natomiast JOY jest męczący, gdyż posiada krótką, cienką rączkę i całkowicie pionowo umieszczony górny FIRE. Rączka ta może ponadto swobodnie obracać się w osi pionowej! Za to nie ma problemów siłowych — nikt nie jest w stanie złamać JOY-a, czego nie można powiedzieć o MATT-cie.

6. Rzut oka do wewnątrz. Jeśli będziecie mieli ochotę zgłębić tajemnice jednego z tych trzech joysticków, z największą uwagą rozbierajcie JOY-a. Jest on skomplikowany co najmniej jak odrzutowiec. Po odkręceniu dwóch śrubek i poruszeniu obudowy wszystko rozsypuje się i potrzeba pół godziny na montaż. Wątpliwości budzi też kabel od FIRE, wpuszczony w obracającą się rączkę. Po kilkunastu obrotach urwie się. Zadziwił nas również boczny FIRE — jest to kilkustopniowy przełącznik wymontowany z bloku przełączników spotykanych np. na skomplikowanych tablicach rozdzielczych.



Matt

Wnętrze MATT-a jest schludne i zaprojektowane z rozmysłem. Rączka wkręcona jest sprężyną głęboko w podstawę. Styki stanowią prawie hermetyczną przestrzeń. Zwierane są gumą przewodzącą, co jest ciekawym i rzadko spotykanym rozwiązaniem.

Podobał nam się również sposób montażu mikroprzełączników przez Joystick Service. Gwarantuje on wytrzymałość i niezawodność dużo większą, niż w wypadku oryginalnych styków.

Podsumowując, za najpełniejszy i najbliższy ideałowi produkt uważamy MATT JOYSTICK! Jego producent, mgr inż. Tadeusz Trojak, oprócz kilku egzemplarzy joysticków (z i bez AUTOFIRE) przysłał nam ciekawe materiały reklamowe oraz kartę wyrobu, normę zakładową oraz wyniki badań przeprowadzonych przez Łódzkie Zakłady Radiowe. Dołączone zostały też pozytywne opinie Zakładów Elektronicznych „ELWRO" oraz P.P. „SOETO".

Joystick MATT uważamy za przykład przemyślanej i celowej koncepcji. Wykonanie nie pozostawia wiele do życzenia.

Ogłaszamy joystick MATT JOYSTICKIEM ROKU 1989!

Przyznajemy również wyróżnienie firmie Joystick Service za działalność na wysokim poziomie. Doceniamy wkład i niezły efekt pracy firmy ELEKTROMECHANIKA, która powinna jeszcze nieco popracować nad swoim produktem.

Pozostaje jedynie mieć nadzieję, że joysticków nie zabraknie. Wszak przybywa komputerów, gier i... dzieci.

P.S. Od 1.10. **Joystick Service** zmienia adres: Nugat 4, Warszawa, pon, śr, pt.

Maciej Pietraś
Marcin Przasnyski

OPERACJE DYSKOWE W SYSTEMIE CP/M Plus

Cz. III — Program formatujący

Informacje zawarte w poprzednich częściach artykułu [1—2], łącznie z opisanymi tam procedurami dyskowymi niskiego poziomu, pozwolą nam przedstawić w tym numerze „Bajtka" program formatujący dyskietki na komputerach Amstrad PCW i CPC.

Zasadniczo, program przedstawiony na listingu 1 został uruchomiony na PCW 8256, ale z pewnymi ograniczeniami może być również używany na CPC 6128. Dotyczy to szczególnie formatów dwustronnych dla stacji 5.25". W wypadku CPC są pewne problemy w dostępie do drugiej strony dyskietki. Ich właściwe rozwiązanie wymaga prawdopodobnie modyfikacji oprogramowania firmowego w ROM'ie, w formie RSX'a, i nie zostało przeze mnie zrobione, m.in. z powodu braku komputera.

Opisywany program formatujący jest dość krótki, dzięki dołączeniu plików FMT.PAS, HEI.PAS i DISK3.SYS, prezentowanych wcześniej. Z tego ostatniego zbioru wykorzystano 5 procedur niskiego poziomu: SetDPB, GetDPB, DD_Format, DD_Check i DD_Write. W pliku FMT.PAS znajduje się opis zdefiniowanych formatów, z jakich może korzystać program. Możliwa jest modyfikacja tego zbioru i uwzględnienie standardów, np. ZX Spectrum lub IBM PC.

Jeśli Czytelnikom prezentowany cykl nie znudzi się, to mamy nadzieję opisać te zmiany, łącznie z programami pozwalającymi na oczyt dyskietek z tych komputerów na Amstradzie. Informacje podane dotychczas pozwalają na napisanie szeregu ciekawych programów, bardzo użytecznych w codziennej działalności. W formie ćwiczenia można polecić następujące tematy:

— program formatujący uszkodzone dyskietki; błędne sektory zaznaczane będą jako oddzielny, niekasowalny zbiór,
— program „odświeżający" — odczytuje, formatuje i zapisuje od nowa każdą ścieżkę,
— program DiscCopy — kopiuje całą dyskietkę,
— program DiscComp — porównuje zawartość dwóch dyskietek, ścieżka po ścieżce, sektor po sektorze,
— procedury zabezpieczające przed kopiowaniem innych programów z danej dyskietki.

Wracając do omawianego programu formatującego, należy zwrócić uwagę na zawartą w nim procedurę **FormatTrack.** W odróżnieniu od podobnej procedury znajdującej się w programie pana Marcina Skóry [3], pozwala na formatowanie ścieżki w dowolnym napędzie po jednej i po drugiej stronie dysku. Istotnie uproszczono wzory na przeplot sektorów (ang. interleave). Procedura **Format** wykonuje formatowanie ze sprawdzeniem, co wydłuża trochę działanie programu, ale daje większą niezawodność.

Uruchomienie programu jest proste i sprowadza się do wprowadzenia, z poziomu systemu operacyjnego, następującego polecenia:

FORMAT D:/n

gdzie *D* to nazwa napędu — A lub B, a *n* — numer formatu. W wypadku zapomnienia parametrów poszczególnych formatów, polecenie

FORMAT ?

podaje te dane.

Opisany program formatujący stanowi doskonałe uzupełnienie do prezentowanej w jednym z poprzednich numerów „Bajtka" stacji dysków 5.25" do Amstrada PCW 8256/8512 [4].

Literatura:

[1] J. Mayer, Operacje dyskowe ... Cz. I, „Bajtek" 1—2/90
[2] J. Mayer, Operacje dyskowe ... Cz. II, „Bajtek" 3—4/90
[3] M. Skóra, „Bajtek" 4/89, str. 22.
[4] J. Mayer, Test stacji ..., „Bajtek" 10/89

Jonasz Mayer

LISTING 1 — Plik FORMAT PAS str. 16

```
Program Format;
{*******************************************************}
{                                                       }
{          Plik FORMAT.PAS    (C) JM   Nov. 20, 1989    }
{                                                       }
{  Program formatuje dyski 3" i 5.25".                  }
{  Obsluguje obie stacje - A: i B:                      }
{                                                       }
{*******************************************************}

{$I DISK3.SYS }    { dolaczenie procedur dyskowych  }
                   { niskiego poziomu               }

procedure FormatTrack (dr : char; track : byte; DPB : DPBrec);
{*******************************************************}
{ Procedura zadaje tablice header zawierajaca informacje }
{ umozliwiajace sformatowanie jednej sciezki. Sciezka jest }
{ formatowana.                                          }
{*******************************************************}
var i, med, headNo, PhTrack  : byte;
begin
  with DPB
  do begin
       case SideMask
       of $00 : begin
                  HeadNo := 0;  PhTrack := track;
                end;
          $01 : begin              { dwustronny: flip sides }
```

```
                if odd(track)
                then HeadNo := 1   { dla nieparz. sciezki }
                else HeadNo := 0;
                PhTrack := track div 2;
              end;
        $82   begin              { dwustronny; up and over }
                if track<40
                then begin HeadNo := 0;  PhTrack := track;
                     end
                else begin HeadNo := 1;  PhTrack :=79-track;
                     end;
              end
      end; { of SideNess }
      med := (PSPT-1) div 2 + 1;           { do przeplotu }
      for i := 1 to PSPT               { petla po sektorach }
      do begin
           header [i,0] := PhTrack; { nr sciezki fizycznej }
           header [i,1] := HeadNo;          { numer glowicy }
           header [i,3] := Hi(SS);  { rozmiar sektora fiz. }
           if odd(i)             { numer sektora fizycznego }
           then header [i,2] := FSN - 1 + (i+1) div 2
           else header [i,2] := FSN - 1 + i div 2 + med;
         end;
      DD_Format (dr,track,$E5);
    end;
end; { of Format Track }

procedure Format (drive : char; DPB : DPBrec; Flag : boolean);
{*************************************************************}
{ Formatuje dysk w napedzie "drive", wg danych z tablicy DPB.}
{ Jesli "Flag" = true, to sprawdzana jest poprawnosc sforma- }
{ towania. W przypadku bledu formatowania program jest prze- }
{ rywany.                                                    }
{*************************************************************}
var
   NoOfTracks, i,j : integer;
begin
  with DPB
  do begin
      if SideNess in [$00,$82]
      then NoOfTracks := TPS
      else NoOfTracks := 2 * TPS;
      for j := 0 to NoOfTracks-1
      do begin
           write (#13,j);
           FormatTrack (drive,j,DPb);
           if not OK
           then begin
                  case error
                  of  0 : write (' Drive not ready');
                      1 : write (' Write protected')
                  else    write (' Format failure (',
                                 error:1,')');
                  end;
                  exit;
                end;
           if flag
           then  for i := 0 to PSPT-1  { sprawdz czy dobrze }
                 do begin              {     sformatowane    }
                      DD_Check (drive,j,i);
                      if not Ok
                      then begin
                             write (' Error in check:',j:2,i:3);
                             exit;
                           end;
                    end;
      end; { of loop over tracks }
    end;
end; { of Format }

procedure WriteBootSector (drive : char; SDS : SDSrec);
{*************************************************************}
{ Zapis skroconej specyfikacji dysku w sektorze zerowym      }
{ zerowej sciezki.                                           }
{*************************************************************}
var i : integer;
begin
  if (SDS.DT in [$00,$03]) and (SDS.SDN in [$00,$81])
  then begin
         Move ( SDS, buf [1], 10);
         for i := 11 to 512 do buf[i] := $E5;
         DD_Write (drive,0,0);
         if not OK
         then write (#7' Error in write');
       end;
end; { Fills Zero Sector Zero Track }

{$I FMT.PAS }

{$I HEI.PAS }

procedure AssignSDS (var SDS : SDSrec; drive : char; nr : byte);
{*************************************************************}
{ procedura przepisuje jedna z krotkich specyfikacji formatu }
{ zadana w tablicach SDS3 i SDS5 do rekordu SDS.             }
{*************************************************************}
var adr : integer;
begin
   case drive
   of 'A','a' : adr := addr (SDS3 [nr]);
      'B','b' : adr := addr (SDS5 [nr]);
   end;
   Move (Mem[adr],SDS,10);
end; { assign SDS }

procedure AssignDPB (var DPB : DPBrec; drive : char; nr : byte);
{*************************************************************}
{ procedura przepisuje jeden z formatow zadanych w tablicach }
{ DPB3 i DPB5 do rekordu DPB.                                }
{*************************************************************}
var adr : integer;
begin
   case drive
   of 'A','a' : adr := addr (DPB3 [nr]);
      'B','b' : adr := addr (DPB5 [nr]);
   end;
   Move (Mem [adr],DPB,27);
end;  { Assign DPB }

{*************************************************************}

var
   ch, drive    : char;     finished : boolean;
   SDS          : SDSrec;   nr : integer;
   DPB, OldDPB  : DPBrec;

begin  { MAIN }

  Init (Drive,nr,'Format','FORMAT');

  GetDPB (drive,OldDPB);    { zapamietanie poprzedniego DPB }
  AssignSDS (SDS,drive,nr);
  AssignDPB (DPB,drive,nr);
  SetDPB (drive,DPB);                     { Zadanie formatu }

  repeat
    Writeln ('Insert new diskette for drive ',drive,':');
    Writeln ('and strike ENTER when ready or EXIT to abort');
    repeat
      read (kbd,ch);
      ch := UpCase(Ch);
    until ch in [#13,#27];
    writeln;
    writeln;
    if ch=#13
    then begin         { formatowanie ze sprawdzeniem }
           Format (drive,DPB,true);
           if OK then WriteBootSector (drive,SDS);
           if OK
           then writeln (#13'Format complete.')
           else writeln (', aborted!');
           writeln;
           write ('Format another (Y/N)?');
           repeat
             read (kbd,ch);
             ch := UpCase(Ch)
           until ch in ['Y','N'];   Finished := (ch='N');
           writeln;
         end
    else Finished := true;
  until Finished;

  SetDPB (drive,OldDPB);       { odtworzenie poprzedniego DPB }
end.

{*************************************************************}
```

Uwagi uzupełniające do testu stacji dysków z nr 10/89

W artykule poświęconym testowi stacji dysków 5.25" do komputera Amstrad PCW 8256, a umieszczonym w numerze 10/89, znalazły się pewne nieścisłości i braki. Dotyczą one między innymi czerwonego przełącznika, widocznego na przedniej ściance, i liczby plików możliwych do przechowywania na dyskietce 5.25".

Ad 1. Opisywana stacja sprzedawana jest w kilku wersjach do wielu komputerów i z różnymi napędami — 40 lub 80 ścieżek. Wyłącznik zasilania znajduje się zawsze na tylnej ściance obudowy, natomiast wyłącznik widziany z przodu, zależnie od wersji, służy do zmiany strony lub do zmiany trybu pracy — 40/80 ścieżek.

Ad 2. W systemie CP/M pojemność dyskietki jest określona przez liczbę jednostek alokacji pamięci i przez liczbę pozycji w katalogu. W opisywanym przypadku tych miejsc w katalogu jest rzeczywiście 256, ale taka liczba plików nie zmieści się na dyskietce, ze względu na zbyt małą liczbę bloków alokacji. Przedstawiony w artykule format jest nie najszczęśliwszą adaptacją oryginalnego formatu dla stacji 80-ścieżkowej. Osobiście używam innych formatów, dostępnych dzięki programowi Format (opis — „Operacje dyskowe... cz. III").

Program ten ułatwia korzystanie z dodatkowej stacji i, razem z programami XTREE.COM (wersja CP/M'owa) oraz IBM.COM (transfer dyskietek IBM-PCW), istotnie zwiększa satysfakcję z dokonanego zakupu.

W prezentowanym tekście nie popierałem stosowania napędu 80-ścieżkowego — niesłusznie. Mimo większej ceny jest on lepszy (większa pojemność — 720-800KB) i nie wymaga, jak napisałem, stosowania dyskietek typu HD, ale tylko dyskietek DD i QD (Quad Density). Te ostatnie, oferowane przez Stilon w cenie 4.5 zł, są niezłej jakości i doskonale nadają się do pracy.

Jonasz Mayer

TEST interface'u dyskowego do komputera Amstrad CPC 464

Wśród potencjalnych Czytelników „Bajtka" znajduje się z pewnością wielu posiadaczy komputera Amstrad CPC 464, niezadowolonych z dotychczasowej tematyki klanu, poświęconego w dużej mierze poważniejszym zastosowaniom, o które trudno przy braku stacji dysków. Osoby te z pewnością pocieszy fakt, że za stosunkowo niewielkie pieniądze mogą nabyć kontroler umożliwiający korzystanie przez 464 z jednego lub dwóch napędów dyskowych.

Obudowa interface'u kontrolera dyskowego ma rozmiary 184x73x21 (mm) i niewielki ciężar. Wykonana jest z metalu i wygląda dość estetycznie. Do komputera kontroler przyłączany jest poprzez 50-stykowe złącze krawędziowe. Wewnątrz obudowy znajduje się kilka układów scalonych i parę elementów biernych. Podstawową rolę odgrywają kości: 765 i EPROM z programem obsługi. Dołączenie napędu odbywa się przez gniazdo DB25. Układ sygnałów odpowiada standardowi stosowanemu w stacjach 5.25" do komputera SpectraVideo SVI — 738, sprzedawanych kiedyś przez CSH. Opis tego standardu przedstawiono w „Bajtku" 3—4/90. Czasami producent preferuje swój własny układ systemu połączeń. W takim wypadku należy to uwzględnić przy zakupie stacji dysków. W nowszych wersjach interface'u stosowane jest złącze krawędziowe, takie same, jak w CPC 6128. Rozwiązanie to ma swoją zaletę, ponieważ można wtedy stosować typowy kabel połączeniowy (standard IBM PC) i przyłączenie dwóch stacji nie wymaga wykonania specjalizowanego rozgałęziacza, który, ze względu na wysoką cenę gniazd i wtyków Canon DB 25, jest kosztowny.

Kontroler łącznie z oprogramowaniem w pamięci EPROM pozwala na współpracę komputera z jedną lub dwiema jednostronnymi stacjami dysków 3" lub 5.25". Nawet w wypadku zastosowania napędów dwustronnych dostęp do drugiej strony jest „ręczny" — odbywa się za pomocą specjalnego przełącznika w obudowie stacji. Nie stosuje się też napędów 80-ścieżkowych. Najczęściej, ze względu na cenę dyskietek, interface sprzedawany jest ze stacją 5.25" (40T) np. tą, która była opisana w „Bajtku" 10/89.

Oprogramowanie

Dzięki oprogramowaniu zawartemu w pamięci typu EPROM, kontroler umożliwia pracę w systemie operacyjnym AMSDOS. Różnice na tym poziomie w stosunku do CPC 6128 są niewielkie i wynikają z mniejszej pamięci operacyjnej 464, wynoszącej 64 KB. Nie ma to żadnego wpływu na gry i nabycie interface'u razem ze stacją istotnie podnosi komfort korzystania z takiego zestawu rekreacyjnego.



Przyznam się szczerze, że mnie bardziej interesowały dydaktyczne możliwości sprzętu, wynikające z podłączenia stacji dysków do komputera. I w tym wypadku nie jesteśmy zawiedzeni. Dyskietka z oprogramowaniem dostarczana łącznie z interface'm zawiera pełny system operacyjny CP/M 2.2. Dzięki programom FileCopy, CopyDisc i ChkDisc możliwa jest wymiana plików między dyskietkami, nawet przy użyciu tylko jednego napędu. Niewielki, w porównaniu z CPC 6128, obszar TPA — 39 KB utrudnia trochę wykorzystanie wszystkich dostępnych w systemie CP/M programów.

W szczególności praca z Turbo Pascalem wymaga pewnej dyscypliny. Po załadowaniu programu dostępny obszar roboczy ma tylko 7KB. Nakłada to warunek na strukturę uruchamianych programów własnych. Należy je podzielić na kawałki tekstów o rozmiarach mniejszych niż 7 KB, a program główny powinien być możliwie najkrótszy i może zawierać tylko instrukcje INCLUDE. Przy zachowaniu takiego reżimu pracy możliwe jest, przy kompilacji na dysk !, tworzenie kodu wynikowego o rozmiarach ponad 30 KB.

Udało mi się m.in. skompilować kilka, nawet długich, programów z dyskietki Turbo Tutora (MicroCalc), a także program BTREE — przykładowa baza danych — z dyskietki Turbo Toolbox. Z powodu małej pamięci nie jest możliwe uruchomienie programu TINST i dlatego dopasowanie Turbo Pascala do pracy z terminalem oferowanym przez 464 musi odbywać się na innym komputerze (np. 6128). Nie powiodło się natomiast uruchomienie procedur graficznych odwołujących się do ROM'u a podobnych do tych, jakie przedstawiono w „Bajtku" 3/89 dla HiSoft Pascala. Myślę, że jest to tylko kwestia czasu i proszę osoby, które to zrobiły, o kontakt z Redakcją.

Podsumowanie

Prezentowany interface pracował bez zarzutu przez ponad pół roku z jedną lub dwiema stacjami (3" i 5.25"). Wygląda estetycznie, jest solidnie zbudowany, w miarę tani (400 tys. zł) i może być polecony wszystkim użytkownikom komputera Amstrad CPC 464. Ze względu na możliwość pracy w systemie operacyjnym CP/M, namawiałbym szkoły lub osoby zainteresowane w poważniejszym wykorzystaniu sprzętu do jego nabycia.

Przypominam też, że standardowy format dyskietek Amstradów to 3 cale i że osoby kupujące razem z interface'em stację 5.25" muszą pamiętać o przeniesieniu potrzebnych programów i zbiorów na ten właśnie format. Niekiedy usługi takie świadczy firma Cieślikowski & ska.

Jonasz Mayer

Testowany interface otrzymaliśmy od firmy Cieślikowski & ska, tel. 29-89-31. Dziękujemy.

—TIMEX— BEZ TAJEMNIC (1)

Nie napisano wiele o tej portugalskiej kopii ZX Spectrum. W dodatku wszystkie te opisy dotyczyły tylko nowych trybów graficznych i gier, które nie działają na Timexie. Pozostało jeszcze wiele spraw nie poruszonych. Głównie dlatego, że potrzebna jest do tego gruntowna wiedza o Timexie — jego budowie i działaniu.

Ulotka dołączona do instrukcji wspomina, że różnice pomiędzy Timexem a Spectrum są minimalne i tylko ok. 5% gier nie działa na Timexie. Nie wyjaśnia ona jednak przyczyny. Czas więc najwyższy, by stworzyć coś na kształt szczegółowej listy różnic pomiędzy tymi komputerami. Nie będzie to prawdopodobnie opis kompletny, gdyż powstał na podstawie testów. Pozostaje więc pole do popisu dla Czytelników „Bajtka".

Do rozpoczęcia testów zmusiło mnie dziwne zachowanie Timexa. Dobrym pretekstem było skończenie się gwarancji.

Już po zdjęciu obudowy rzucają się w oczy pierwsze poważne różnice. Timex został zaprojektowany z wyjątkowym rozmachem i nie jest to niewolnicza kopia. Wspólne ze Spectrum pozostały tylko niektóre oporniki, kwarce, mikroprocesor i modulator TV. Oprócz tego są już tylko kości **RAM**, **EPROM** w miejsce oryginalnego **ROM-u**, dwa układy **LS157** zamiast czterech, układ **LS244** do obsługi joysticka i oczywiście całkiem nowa płytka drukowana.

Wiadomo, że Timex ma bogatsze niż Spectrum możliwości graficzne (szkoda, że tak rzadko wykorzystywane). Zawdzięcza on to nowej **ULI**. Została ona wykonana na zamówienie w obudowie kwadratowej (4x17 nóżek). Przejęła wszystkie funkcje **ULI** ze Spectrum i dodatkowo trzech-czterech układów scalonych. Oprócz nowej grafiki ma także inne funkcje: może wyłączyć spod kontroli mikroprocesora część lub całą pamięć (**ROM** i **RAM**), może wyłączyć też generowany przez siebie co 1/50 sekundy sygnał przerwania maskowalnego.

Zanim rozpocznę szczegółową listę różnic, ucieszę właścicieli Timexów informacją, że niektóre egzemplarze mają fabrycznie zamontowane **80 KB RAM**. Zawierają one **16 KB** tzw. **VIDEO RAM-u** i dwa banki po **32 KB RAM** każdy. Niestety, nie można programowo wykorzystać drugiego banku **32 KB** bez ingerencji w obwód elektryczny płytki drukowanej. Pierwsze próby wykazały, że przeróbka jest możliwa i stosunkowo prosta; przedstawimy ją wkrótce w „Bajtku".

1. Tryby graficzne

Wszystkie tryby graficzne uzyskuje się przez rozkaz **OUT 255,nr.** Jednocześnie wartość aktualnego trybu graficznego można odczytać np. **PRINT IN 255.** Po włączeniu Timexa zawsze ustawiony jest tryb grafiki Spectrum — zawartość portu **255** wynosi **O.** Zapewnia to drobna i jedyna zmiana w **ROM-ie**. Pod adresami **4762** i **4763** zamiast wartości 10,12 (#OA,#OC) oznaczających adres skoku do procedury drukującej napis „© 1982 Sinclair..." jest adres skoku na koniec interpretera Basic-a w **ROM-ie** po adres **14446**. Są tam trzy nowe rozkazy:

```
14446 #386E #D3 #FF      OUT (#FF).A    ; wysłanie zawartości
14448 #3870 #CD #OA OC   CALL #OCOA     ; A na port 255. wydruk
14451 #3873 #C9          RET            ; „©". powrót pod 4764
```

Zainteresowanych samodzielną obsługą rozszerzonej grafiki odsyłam do artykułu w nr **1/88 „InforMika"**. Przypominam, że są cztery podstawowe tryby: [**O** — tryb Spectrum; komórki pamięci od **16384** do **22527** **(#4000—#57FF)** określają treść ekranu, od **22528** do **23295** **(#5800—#5AFF)** — atrybuty,

1 — tryb podobny do poprzedniego, ale treść ekranu rozpoczyna się od **24576** (#6000), a nie **16384** (atrybuty od **30720** (#7800),

2 — adresy treści ekranu są takie jak w trybie Spectrum, ale każdemu bajtowi treści odpowiada bajt atrybutów znajdujący się **8192** bajty dalej,

6 — tryb podwojonej rozdzielczości zostanie omówiony później,

Przeanalizujmy teraz poszczególne bity portu 255:

OUT 255, 7 6 5 4 3 2 1 0

bit 0 — zapalenie go powoduje, że **ULA** konstruuje obraz TV, pobierając informacje z adresów **#6000—#7AFF** zamiast **#4000—#5AFF**

bit 1 — zapalenie powoduje, że **ULA** pobiera informacje o atrybutach z adresów **#6000—#77FF** zamiast **#5800—#5AFF**

bit 2 — zapalenie go włącza tryb podwojonej rozdzielczości (64 kolumny x 24 wiersze tekstu — 512x192 punktów). W kolumnach parzystych umieszczana jest treść ekranu, a w kolumnach nieparzystych zawartość atrybutów ekranu

bity 3,4 i 5 określają kolor tuszu wspólny dla całego ekranu:

	bit 5	bit 4	bit 3	kolor
0	0	0	0	czarny
1	0	0	1	niebieski
2	0	1	0	czerwony
3	0	1	1	fioletowy
4	1	0	0	zielony
5	1	0	1	jasnoniebieski
6	1	1	0	żółty
7	1	1	1	biały

kombinacje bitów 0, 1 i 2 dają ciekawe efekty:

OUT 255,6 — pełny tryb podwojonej rozdzielczości kolumny parzyste na podstawie komórek **#4000—#57FF**, nieparzyste na podstawie **#6600—#77FF**. Kolory określają bity **3, 4** i **5**, czyli tusz jest czarny, a tło białe,

bit 6 — zapalenie go powoduje wyłączenie generowania przerwań maskowalnych przez układ **ULA**. Przekonuje o tym prosty programik:

10 OUT 255,64
20 OUT 255,0

który wykona się bez zakłóceń. Wystarczy teraz dopisać:

15 PAUSE 0

aby komputer się „zawiesił". Powodem jest wyłączenie przerwań uniemożliwiające odczytywanie klawiatury i przejście z linii **15** do linii **20** i zakończenie programu,

bit 7 — nie używany.

cdn.

Marek Sawicki

GRAFIKA TIMEXA

Dotychczas mało się mówiło, a raczej dużo milczało o rozszerzonych możliwościach graficznych Timexa.

Przy okazji początku cyklu o jego wnętrzu proponujemy dwie procedurki maszynowe, które pomogą (usprawnią, poprawią itp.) w tworzeniu grafiki podwyższonej rozdzielczości.

GRAF:

Program ten powoduje przeniesienie ekranu w rozdzielczości **256 * 192** na ekan **512 * 192** „rozciągając" każdy bajt. Tak przeniesiona zawartość może być podretuszowana, niestety na razie tylko przy pomocy **PLOT**. Jestem w trakcie opracowywania komend **DRAW**, **CIRCLE** do podwyższonej rozdzielczości.

Po wpisaniu tekstu źródłowego pod **GENS** i skompilowaniu lub wpisaniu programu ładującego Basic-u demonstracja nastąpi po uruchomieniu programu:

```
10 CLEAR 24575: OUT 255,62: LIST
20 RANDOMIZE USR 5e4: PAUSE 0: STOP
```

KOLOR:

Ten program wykorzystuje fakt przypisania każdemu bajtowi osobnych atrybutów. Po umieszczeniu kodu wynikowego w pamięci proszę uruchomić (np. po **NEW**) program:

```
10 CLEAR 24575: OUT 255,2: LIST
20 RANDOMIZE USR 5e4: PAUSE 1: GO TO
   20
```

```
kolor

:NIC

   1 CLEAR 4E4
   2 LET S=0: FOR X=0 TO 50: REA
D A: LET S=S+A: POKE 5E4+X,A: NE
XT X
   3 IF S<>5053 THEN PRINT "POPR
AW !": LIST 10: STOP
   4 SAVE "KOLOR FOR TMX"CODE 5E
4,51
  10 DATA 58,131,195,60,230,7,50
,131,195,6,8,33,0,96,17,1,96,197
,1,0,1,119
  20 DATA 237,176,60,230,7,193,1
6,243,33,0,96,17,0,104,1,0,8,229
,197,237,176,193
  30 DATA 225,17,0,112,237,176,2
01


   10            ORG   50000
   20            LD    A,(AD1)
   30            INC   A
   40            AND   7
   50            LD    (AD1),A
   60            LD    B,8
   70            LD    HL,24576
   80            LD    DE,24577
   90  DD        PUSH  BC
  100            LD    BC,256
  110            LD    (HL),A
  120            LDIR
  130            INC   A
  140            AND   7
  150            POP   BC
  160            DJNZ  DD
  170            LD    HL,24576
  180            LD    DE,26624
  190            LD    BC,2048
  200            PUSH  HL
  210            PUSH  BC
  220            LDIR
  230            POP   BC
  240            POP   HL
  250            LD    DE,28672
  260            LDIR
  270            RET
  280  AD1       DEFB  7
```

Uwaga! Nie sprawdzałem, jak powyższe programy zachowują się w wypadku zainstalowania 80 KB przełączanego OUT 255,32. Nie powinno być kłopotów, gdyż wystarczy tu zmiana jednego bitu, lecz gdyby wystąpiły trudności, proszę o powiadomienie.

Piotr Borowski

```
graf

   1 CLEAR 4e4
   2 LET S=0: RESTORE : FOR X=0
TO 68: READ A: POKE 5E4+X,A: LET
 S=S+A: NEXT X
   3 IF S<>7316 THEN PRINT "POPR
AW !": LIST 10: STOP
   4 SAVE "PROG"CODE 5E4,69
  10 DATA 243,33,0,64,17,0,96,1,
0,24,197,6,4,126,50,149,195,58,1
49,195,31,50
  20 DATA 149,195,26,205,137,195
,18,16,242,6,4,58,149,195,31,50,
149,195,126,205,137,195
  30 DATA 119,16,242,193,35,19,1
1,120,177,32,211,251,201,56,5,31
,31,230,63,201,31,31
  40 DATA 246,192,201


   10            ORG   50000
   20            LD    HL,16384
   30            LD    DE,24576
   40            LD    BC,6144
   50  POCZ      PUSH  BC
   60            LD    B,4
   70            LD    A,(HL)
   80            LD    (AD1),A
   90  AA        LD    A,(AD1)
  100            RRA
  110            LD    (AD1),A
  120            LD    A,(DE)
  130            CALL  PODPRO
  140            LD    (DE),A
  150            DJNZ  AA
  160            LD    B,4
  170  Q1        LD    A,(AD1)
  180            RRA
  190            LD    (AD1),A
  200            LD    A,(HL)
  210            CALL  PODPRO
  220            LD    (HL),A
  230            DJNZ  Q1
  240            POP   BC
  250            INC   HL
  260            INC   DE
  270            DEC   BC
  280            LD    A,B
  290            OR    C
  300            JR    NZ,POCZ
  310            RET
  320  PODPRO    JR    C,BB
  330            RRA
  340            RRA
  350            AND   %00111111
  360            RET
  370  BB        RRA
  380            RRA
  390            OR    %11000000
  400            RET
  410  AD1       DEFB  0
```

CP/M start!

Drodzy Spectrumowcy! Klepiecie w swą Trumnę kolejny rok, dokupujecie do niej, co się da, gracie, oglądacie dema, piszecie programy. Większość z Was ma już stację dysków. Pomyślcie, co robić dalej. Nie proponuję Wam jeszcze IBM-a, choć muszę powiedzieć, że nie byłaby to bigamia.

Proponuję jednak CP/M! To wielki krok w kierunku IBM-ów, wielki krok ku profesjonalizmowi. Nie tracicie nic ze starych nawyków i przyjemności, a macie okazję zacząć naprawdę działać.

Do rzeczy. Co to jest CP/M? Jest to system operacyjny, język komunikacji użytkownika z komputerem. Powstał już dawno temu w firmie Digital Research, mimo to ma się wciąż dobrze.

Mówicie, że CP/M na Spectrum to odrobinę sztuczne? I tak, i nie. Do „małego" programowania, grania itp. wystarczy TOS i pamięć Spectruma. 64K pamięci stacji leży odłogiem! To na niej pracuje CP/M pozostawiając Spectrum prawie w spokoju. Wasza Trumienka staje się po prostu nieco inteligentną klawiaturą z głośniczkiem.

Robimy więc ten historyczny krok. W pudełku Waszej stacji powinna znajdować się co najmniej jedna dyskietka z CP/M-em. Na stronie A oznaczona jest jako „CP/M Terminal", na drugiej zaś jako „CP/M System Disk". Pierwsza strona potrzebna jest do zaemulowania terminala systemu operacyjnego na Spectrum. Co to oznacza?

Spectrum zostaje „przestawione" na posłuszeństwo stacji i odpowiada tylko za obsługę obrazu, klawiatury, głośniczka i zegara systemowego. Wykonuje to niewielki program maszynowy, znajdujący się na stronie A wspomnianej dyskietki, ładowany automatycznie.

Należy mocno zaznaczyć, że terminali takich jest kilka, począwszy od oryginalnego Timexowskiego, skończywszy na Polbritowskim, zawierającym polskie znaki i napisy systemowe. Gwoli ogólnej korzyści, opracowaliśmy terminal pośredni, rozbudowany o funkcje realizujące grafikę, dźwięk, RAM-dysk oraz obsługę drukarki przez interface „Centronics Microface". Jest on niezbędny dla Turbo Pascala, o czym za chwilę.

Restartując więc TOS ze strony A dyskietki, o której wciąż mówimy, instalujemy terminal, który prosi nas o włożenie do napędu A: dyskietki systemowej CP/M-u i wciśnięcie RESET w stacji. Po odwróceniu dyskietki i RESET instaluje się CP/M 2.2.

Zgłasza się on w 64-znakowym trybie graficzno-tekstowym, tak jak zgłaszają się systemy operacyjne na całym świecie, tzn. podając oznaczenie aktualnego napędu roboczego (zwykle A:) i znaku zachęty „›". Zaraz za nim miga kursor, prosząc nas o rozkazy.

W tym momencie zaczynamy nabierać nawyków, które pomogą nam w przyszłości w pracy na IBM-ach (wciąż w to wierzę!). Podobnie jak w MS-DOS-ie, tak w CP/M-ie dysponujemy zestawem komend (poleceń) systemu, za pomocą których działamy. Wszelkie operacje odbywają się z dyskietką, tzn. nie wolno jej usuwać, gdyż każda niemal komenda powoduje w pierwszej kolejności „grzebnięcie" po dysku.

UWAGA! Zmieniając dyskietkę należy zainicjować system operacyjny wciskając RESET w stacji lub prościej — kombinację klawiszy CTRL-C. Ten CTRL to EXTENDED MODE, moi drodzy.

Pierwszą podstawową, najczęściej używaną, najwspanialszą komendą systemu jest DIR. Powoduje ona wyświetlenie katalogu (directory) dyskietki, tzn. spisu znajdujących się tam plików. Pliki z rozszerzeniem. COM — to programy przeznaczone do bezpośredniego uruchomienia przez podanie nazwy (głównego członu). Mogą być to też programy obsługi nierezydentnych komend systemu, co na jedno wychodzi.

Z innych komend rezydentnych wymienić można TYPE — wypisanie (np. na ekran) zawartości dowolnego pliku. Oczywiście po słowie TYPE należy podać jego całą nazwę.

ERA powoduje skasowanie wybranego pliku lub ich grupy. Dopuszczalne jest używanie wzorców kombinowanych z * i ?, tak jak w TOS-ie. * zastępuje dowolny ciąg znaków, a ? dowolny jeden znak.

Na razie nie potrzebujecie wiedzieć więcej. Należy natomiast rzec, co można z pomocą CP/M-u zdziałać.

A można wiele. Przede wszystkim programować, czyli przetwarzać. Ale na poziomie, jeśli można tak powiedzieć, światowym. Co to dokładnie znaczy, poznacie sami. Korzyści są ogromne.

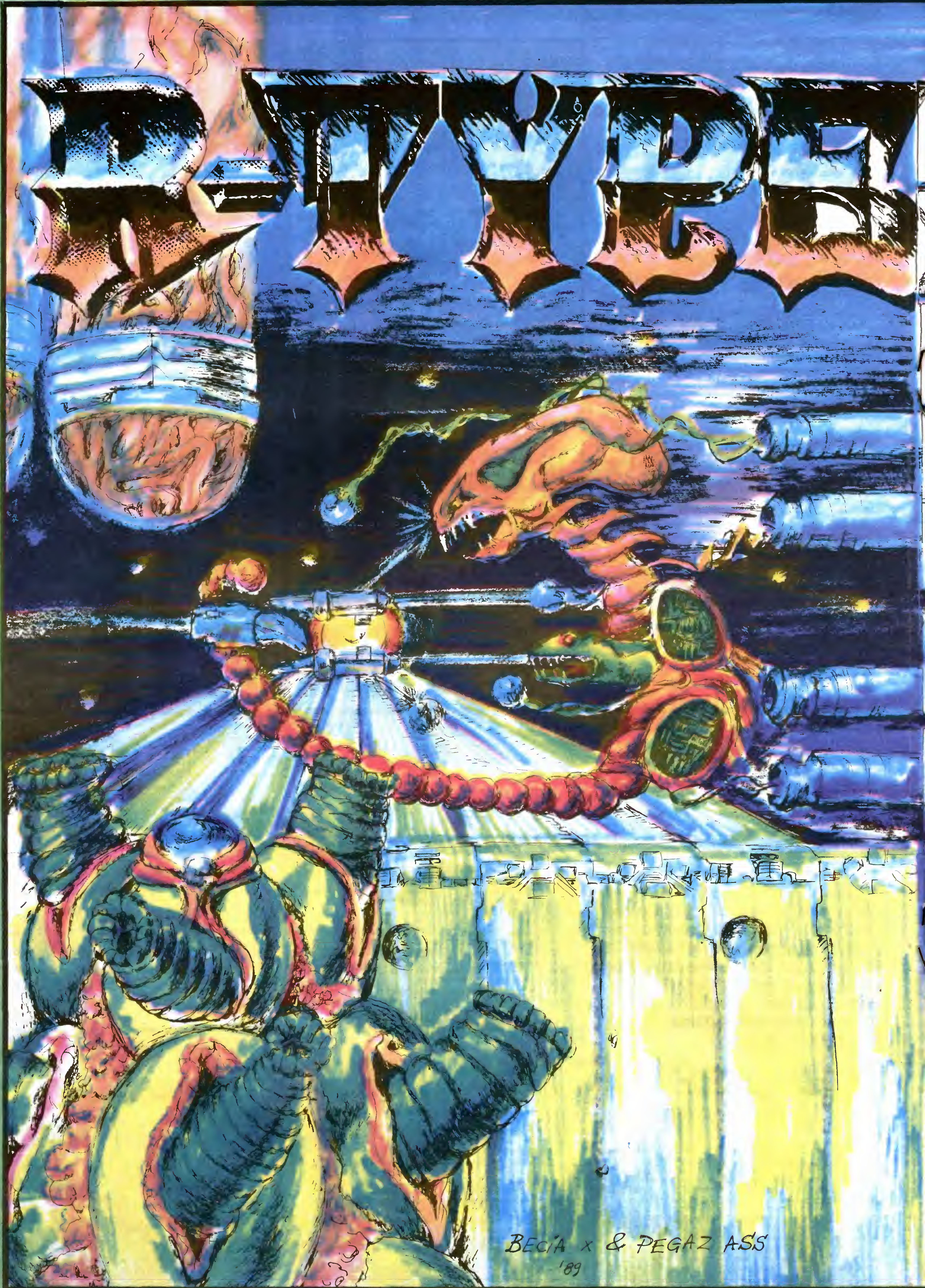
W niewielkiej grupie maniaków opracowaliśmy instalację Turbo Pascala 3.0 firmy Borland wraz z ogromnym środowiskiem, bibliotekami procedur itp. Przy okazji powstał bardzo rozbudowany terminal. W sumie jest to potężne narzędzie programistyczne, pozwalające na wszystko, co w Pascalu możliwe. Co więcej, dzięki możliwościom Spectrum mamy i grafikę, okna, dźwięk też przez AY-greka, obsługę drukarki i moc innych nowości. Co to oznacza, nie muszę przekonywać. Rewolucję.

Zainstalowanie Turbo Pascala możliwe jest poprzez redakcję, jak również poprzez DH „Sezam" w Warszawie, który ma na to licencję. Ostrzegamy przed nie działającymi wersjami giełdowymi, które pozbawione są bibliotek i uruchomione, choć nie zainstalowane, na terminalu Odry.

Czy to, o czym mówimy, nie jest ułudą, półfikcją? Na świecie CP/M egzystuje na nieczęsto spotykanych Amstradach PCW i CPC 6128. Poza tym odchodzi się nawet od DOS-u. Czy na tym tle nie wyglądam śmiesznie zachwycając się dopiero co odkrytym CP/M-em na Spectrum? Chyba nie. Przeciętny użytkownik Spectrum, jak uważam, nie kupi sobie IBM-a przez najbliższe trzy lata. Co najwyżej posługuje się nim w pracy. To dla niego duża dogodność pracować na Spectrum w jakimś wyższym standardzie. Poza tym mamy niezliczonych fanatyków Spectrum, którzy prawie nie wiedzą, co ze sobą zrobić. Dla nich CP/M, Turbo Pascal to nowe narzędzia do wyżycia się, a i nauczenia czegoś naprawdę istotnego i pożytecznego.

Cykl artykułów o CP/M-ie będziemy kontynuowali. Czekamy na listy.

Marcin Przasnyski


R-TYPE
BECIA x & PEGAZ ASS
'89

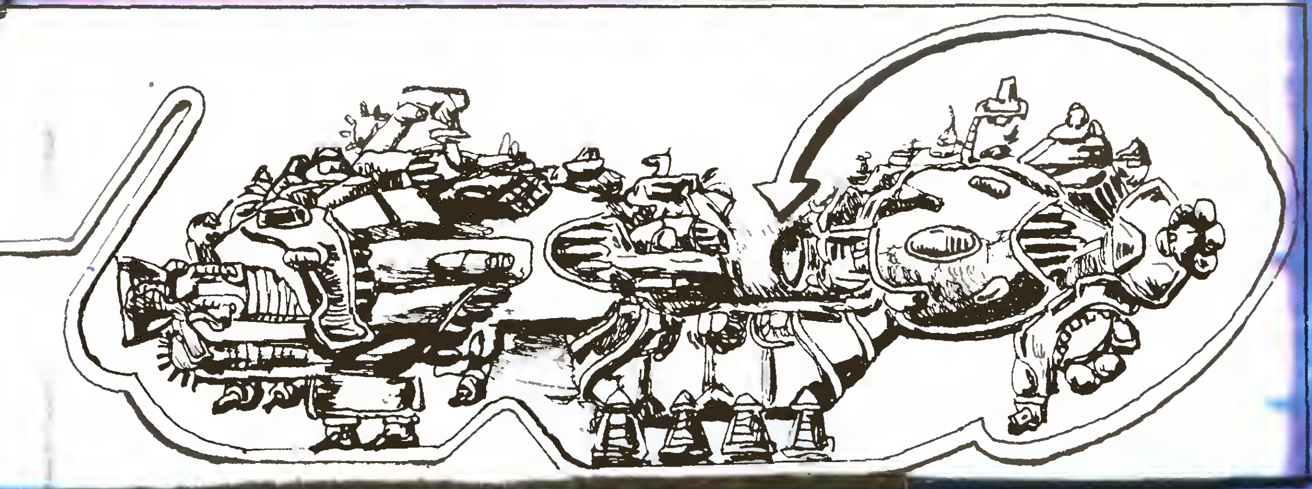


1

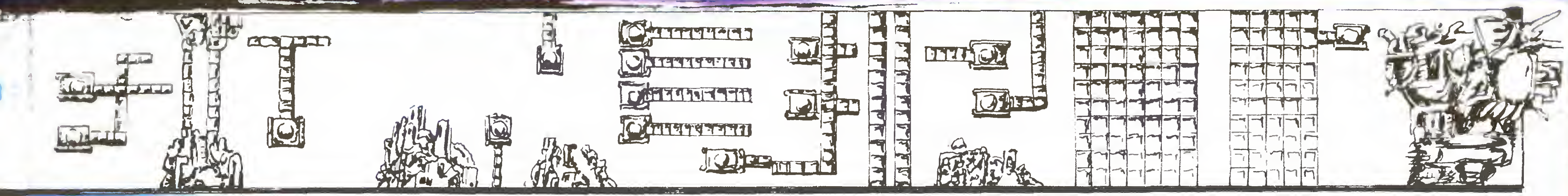




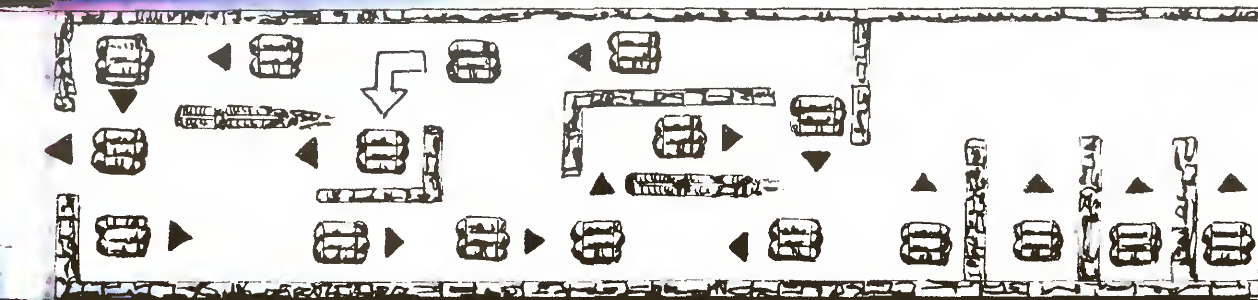
2



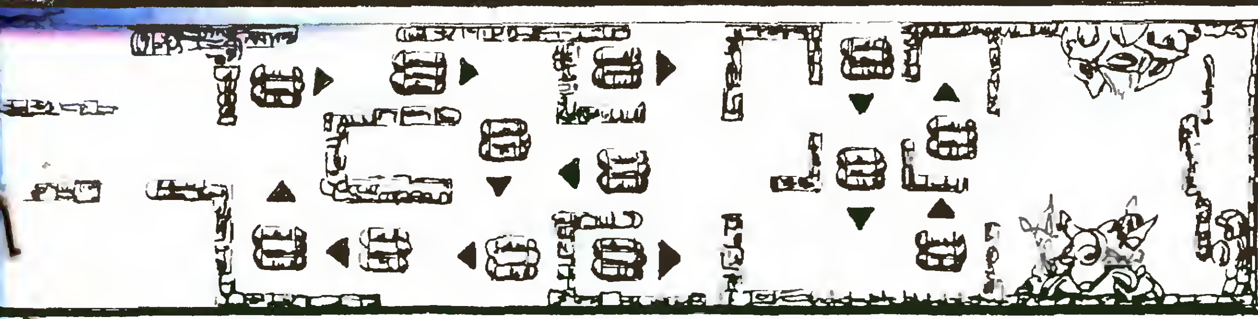
3



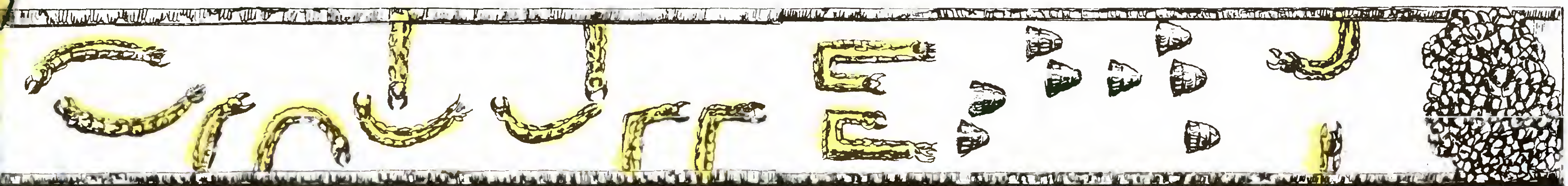
4

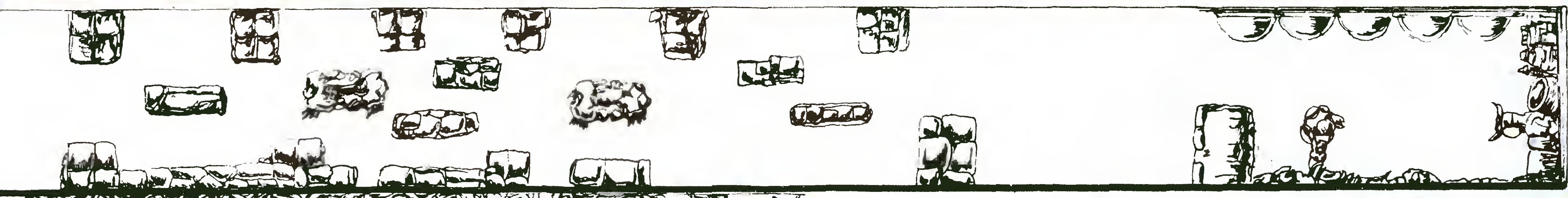


5



6





7



8

10 BAJTKOWA LISTA PRZEBOJÓW 5–6/90

Gry zmieniają się jak w kalejdoskopie. Stare odchodzą, przychodzą nowe i najnowsze, niektóre powracają. W tym miesiącu, a właściwie dwóch miesiącach królują Nietykalni, czyli Untouchables firmy Ocean, gra oparta ściśle na scenariuszu znanego filmu Briana De Palmy. Pojawił się też Test Drive II, również na Spectrum, doskonała symulacja firmy Accolade. Dalej tylko niewielkie przegrupowania.

			ATARI	AMSTRAD	COMMODORE	SPECTRUM
1	THE UNTOUCHABLES	!	<	>	×	×
2	WAR IN MIDDLE EARTH	↑	<	>	×	>
3	HOSTAGES	↑	<	×	×	
4	INDIANA JONES	↓	<	>	×	×
5	ROBBO	↑	>			
6	TEST DRIVE II	!	<	×	×	×
7	RED HEAT	↓	<	>	×	>
8	BATMAN	↓	<	>	×	×
9	TOTAL ECLIPSE	↑	<	>	×	>
10	CABAL	↓	<	>	×	×





KILLED UNTIL DEAD

Zgodnie z tytułem, Twoim zadaniem będzie zabicie Go. Nikt z nas nie wie, kim jest ON. Nie wiemy także, dlaczego mamy GO zabić. Pewne jest tylko, że ON nie pożyje długo, gdy zasiądziemy przed komputerem. Gra ta może wykształcić nowego wspaniałego Sherlocka Holmesa, ale równie dobrze i następcę Kuby Rozpruwacza.

Akcja toczy się w starym, ponurym dworze. Cały dom naszpikowany jest kamerami, a podsłuchy założono nawet na klatce schodowej i w ubikacji. Zainstalowano ten sprzęt na rozkaz Lorda Glanse'a, który obawiał się zamachu na swoje życie. Oczywiście ponury morderca wykonał swój plan, mimo tych wszystkich zabezpieczeń, i Lorda znaleziono pewnego słonecznego dnia bez głowy, która leżała kilka metrów dalej. Pamiętaj, że gdy nie będziesz ostrożny i uważny, to Twojej głowy prawdopodobnie wcale nie znajdą.

Jesteś samotnym detektywem i w misji odnalezienia zabójcy będziesz dowodził wyłącznie podsłuchami. Wbrew pozorom urządzenia te są bardzo przydatne — dzięki nim można rozmawiać nie ruszając się z miejsca. Wszystkie ważne rzeczy zawsze możesz nagrać i odtworzyć na magnetowidzie.

Oprócz podsłuchów posiadasz także telefon, przez który rozmawiasz ze wszystkimi mieszkańcami domu. Są oni bardzo źle usposobieni do Ciebie i aby wyciągnąć od nich informacje, należy zadać kluczowe pytanie. Stają się po nim bardzo mili i skorzy do współpracy. Posiadasz także notatnik, w którym zapisywane są wszystkie rozmowy, przypuszczalne miejsca i godziny spotkań domowników itp.

Na stole leży teczka z aktami personalnymi, gdzie znajdziesz mnóstwo bardzo ciekawych i przydatnych informacji.

Postacie zamieszkujące dom spiskują za Twoimi plecami, umawiając się na różne tajemnicze spotkania. Ich rozmowy są czasami bardzo pouczające i wyjaśniają dużo rzeczy, np.: skąd wzięła się postać grasująca z nożem po trawniku lub kim był wisielec widoczny z frontowego okna. Niekiedy dialogi toczą się na tematy nudne i nie wnoszą nic nowego do sprawy.

Aby dowiedzieć się o miejscu i godzinie spotkania, należy włamać się do pokoju podejrzanego. Niestety, pilnuje go strażnik, który zadaje Ci pytanie. Trafna odpowiedź procentuje informacjami zebranymi w pokoju, a niewłaściwa — zamknięciem wejścia do niego. Tak więc zadanie nie jest łatwe, choć tak naprawdę, to rozwiązywaliśmy już trudniejsze...

W sumie, program ten dobry jest na długie, deszczowe wieczory. Możliwość wyboru różnych scenariuszy, a nawet praktyki, jeszcze bardziej go uatrakcyjnia. Wspaniałe tło w postaci muchożernych kwiatów, zabójców w pończochach rozwesela smutny wieczór. Nigdy dotąd nie zrobiono programu tak wspaniale oddającego atmosferę zbrodni i oszustwa jak ten.

Firma: ACCOLADE
Komputer: Commodore 64, Spectrum

Luke

JAK REKLAMOWAĆ SIĘ W BAJTKU I NIE TYLKO W BAJTKU

Spółdzielnia „Bajtek" oferuje Państwu kompleksową obsługę reklamową na łamach wydawanych przez nią tytułów:

„Bajtek" — obecny na rynku od ponad pięciu lat
„Moje Atari" — poświęcony komputerom Atari
„Top Secret" — magazyn fanów gier komputerowych, dla wszystkich graczy niezależnie od typu komputera

Ceny ogłoszeń:
1 cm^2 — 10 tys. zł
strona — 10 mln zł
kolor — odpowiednio 100% drożej

minimalny format ogłoszenia — 20 cm^2
kolejne formaty — 40, 80, 160, 320, 640 cm^2
Oferujemy też reklamy polegające na testach użytkowych sprzętu oraz programów — warunki do uzgodnienia.

Reklamy przyjmuje: **Spółdzielnia „Bajtek" ul. Wspólna 61 00-687 Warszawa**



INTERNATIONAL KARATE

„Podążał tą drogą od lat. Zawsze gdy wyruszał z wiadomościami do klasztoru, przechodził przez te niezmierzone przestrzenie górskich ścieżek. I za każdym razem zbaczał z drogi, by ujrzeć jezioro Hiuna.

Był już coraz bliżej. Piątym zmysłem wyczuwał i umiejscawiał jezioro. Rzadkie powiewy przynosiły z sobą zapach wody. Cała okoliczna przyroda wyglądała niebiańsko. Szczyty gór rzucały cienie przypominające boskie stworzenia. Wszędobylska zielona trawa stanowiła wspaniałe podłoże dla nóg. Olbrzymie jaskinie mówiły o nieograniczonej sile natury. Wszystkie te elementy współgrały ze sobą tworząc krajobraz gór Hio.

Niespodziewany pisk porozumiewającej się pary świstaków przypominał mu o Agawie. Kochał ją od dawna, miłością młodzieńca niedoświadczonego jeszcze przez los. Wciąż obawiał się jednak, czy ona odzwajemnia to jego czyste uczucie. Te częste uśmieszki, półsłówka, jakie do niego kierowała, mogły równie dobrze oznaczać uwielbienie, jak i nic nie znaczący flirt. Najbardziej niebezpieczni jednak byli dla niego bracia Han, którzy również zabiegali o jej względy, za co nienawidził ich z całego serca. Marzył o tym, aby Agawa wyznała mu swą miłość, na co sam nie mógł się dotąd zdecydować, lub też odrzuciła go na zawsze. Ciągła niepewność męczyła go coraz bardziej, stając się nie do zniesienia.

Olbrzymia tafla jasnoczerwonego słońca chowała się powoli za wierzchołkami gór. Wiedział jednak, że zobaczy ją za chwilę znowu — każdy zachód było wspaniale widać z brzegu jeziora.

Gdy ujrzał je, zadrżał ze zgrozy. Niedowierzanie po chwili przemieniło się w dziką nienawiść. To, co zobaczył, było dla niego straszne. Jego dwaj najwięksi wrogowie — Han i jego młodszy brat Han-go — stali nad jeziorem łapiąc rękami ryby, które garnęły się do nich z ekstazą godną lepszej sprawy.

Nie myślał ani sekundy, wszystko, co robił, kierowane było tylko przez jego podświadomość. Z okrzykiem „Banzai" zaczął zbiegać w dół. Natychmiast obrócili się, gotowi do obrony przed niespodziewanym niebezpieczeństwem. Przyjęli typową postawę dla południowych odmian karate. Nogi, które były ich najgroźniejszą bronią, rozstawili szeroko.

Od początku nie mieli jednak szans. Zaskoczenie i wściekłość napastnika wraz z dużymi umiejętnościami nie dawały im nadziei. Wyskok z obu nóg, olbrzymie salto w powietrzu i pewne lądowanie tuż obok nich, pokazało im jego klasę. Mieli jednak zamiar walczyć do końca, gdyż najstraszniejszą myślą dla każdego japońskiego wojownika jest śmierć bez walki. Han i Han-go nie stanowili wyjątku.

Krążył wokół nich powoli i bardzo uważnie. Wiedział doskonale, że pokonać dwóch przeciwników naraz jest prawie niemożliwe. Musiał zaczekać na ich atak i dobrze go skontrować. To była jego jedyna szansa...

Pierwszy ruszył Han-go. Z niesamowitą szybkością podskoczył do przeciwnika i wyprowadził doskonały cios pięścią w głowę. Popełnił jednak dwa błędy: nie docenił przeciwnika i zaatakował bez wsparcia brata. Karą za to było uderzenie nogi oraz podcięcie, jakie po chwili otrzymał. Han-go przewrócił się, tracąc chwilowo przytomność.

Jego brat Han uderzył szybko, bardzo szybko a mimo to za wolno. Dwa natychmiastowe kopnięcia mierzone w tułów nie doszły celu. Han miał jeszcze tyle czasu, by dwoma sprawnymi i doskonałymi blokami sparować dwie kontry przeciwnika. Natychmiast też odskoczył na bezpieczną odległość. Jego brat Han-go już odzyskiwał przytomność.

Po nieudanym początku postanowili zastosować inną taktykę, obchodzili przeciwnika tak, że cały czas znajdował się między nimi. Gdy zbliżyli się na bardzo małą odległość, podskoczyli szybko i... To był już ich trzeci błąd. Zobaczyli tylko wspaniały podskok przeciwnika i jednocześnie ciosy nogami na obie strony. Nie zdążyli wykonać najmniejszego ruchu, nic nie mogło już im pomóc. Poczuli straszliwe..."

(...)

Na tym kończy się urwany zapis w księdze magicznej klasztoru mistrza Kio-lao, położonego w górach Hio. Nikt nie wie, jak skończyła się sławna walka braci Han z mistrzem tego klasztoru Tienem. Kto przeżył, a kto zginął? Na to pytanie może odpowiedzieć już tylko błękitna toń jeziora Hiuna.

Każdy z nas ma jednak szansę dokończyć walkę karateków, używając do tego celu komputera z załadowaną do niego grą International Karate. Do dyspozycji mamy wiele różnorodnych ciosów, bloków, widowiskowych wyskoków i podcięć. Pamiętać jednak należy o jednej starej, ale wciąż jarej zasadzie: „gdzie dwóch się bije, tam trzeci... obrywa".

Firma: Melbourne House
Komputer: Spectrum 48/128, Commodore 64, Atari XL/XE, Atari ST, Amiga

Luke

S.O.S.

Poszukuję następujących gier na komputer Timex 2048: BARBARIAN, WORLD KARATE, CHAMPIONSHIP, ARCHON, EXOLON. Odstąpię za nie WIZBALL, REBEL STAR, INSPECTOR 6 lub nieśmiertelności. **Marek Łapczyński, ul. Czajewicza 14/5, 05–500 Piaseczno**

Jestem posiadaczem Atari 130XE. Nie wiem jak przejść komnatę z komputerem w UNIVERSAL HERO. Nie wiem też co zrobić w grze JOE BLADE. Proszę o mapę tej gry. Będę bardzo wdzięczny. **Paweł Góralski, ul. Grochowska 202/55, 04–357 Warszawa**

Poszukuję opisów do gier MAGNETRON, AVENGER, BACK TO THE FUTURE, MAX HEADROOM I i II, ROGUE TROPPER i ROBOT MESSIAH. W zamian wiele opisów i nieśmiertelności. **Marcin Prokop, ul. Szaserów 102/7, 04–335 Warszawa**

Poszukuję pierwszych numerów Bajtka (czarnobiałych) i schematów udoskonaleń Spectrum. Szukam też schematów komputerów Spectrum, Timex 2048, Commodore (16, 116, +4, 64, 128) oraz Atari 800XL. **Jacek Rubinkowski, ul. Orlińskiego 3/20 09–400 Płock**

Posiadam komputer Atari 800XL z magnetofonem. Proszę o pomoc w grze BOULDER DASH. Nie wiem jak przejść komnatę „K" oraz jak uzyskać nieśmiertelność w tej grze. Proszę również o opisy gier ZORRO i RAID OVER MOSCOW. Liczę na pomoc czytelników. **Monika Bartz, ul. Korzenna 11a m 8 81-587 Gdynia-Karwiny 1.**

Pilnie szukam opisów gier na Atari 800XL: SUBMARINE COMMANDER i SILENT SERVICE. **Szymon Marcinek, Bukowiec Górny 8b 64–141, woj. leszczyńskie**

W związku tym, że mam Atari 800XL proszę o przysłanie mi dokładnych opisów do gier AGENT U.S.A., BLUE MAX, MAX 2001 i CLAIN JUMPER. **Rafał Orłowski, ul. A. Mickiewicza 8 m 12 06-406 Opinogóra**

Mam komputer Atari 65XE. Proszę o dokładny plan i opis gry SILENT SERVICE. W zamian 10 innych opisów gier. **Andrzej Dziewian, ul. Wilka 8a m9 41–705 Ruda Śląska 5 woj. katowickie**

Poszukuję programów FAST HACK'EM lub PRINT MASTER. Proszę także o przysłanie mi gry KUBA ROZPRUWACZ, która została zamieszczona w BAJTKU. Jestem posiadaczem C–64 z magnetofonem. **Borysław Panlewicz, ul. Malczewskiego 13 m 20 43–300 Bielsko Biała**

Za kilkanaście gier na Atari 65XE oferuję programy na komputer Amstrad CPC 464. **Michał Kardasiński, ul. Klonowa 46 m 31 25–538 Kielce**

Jestem posiadaczem Commodore 64 z magnetofonem. Poszukuję gry FLIGHT SIMULATOR II i opisów POLICE ACADEMY i THE LAST NINJA. **Michał Zakrzewski, ul. 1000-lecia 6a m 8 86–320 Łasin**

R – TYPE

opis do mapy

Nie tak dawno, ale i niezbyt niedawno, daleko w Kosmosie, chociaż nawet całkiem blisko, żył ludek niewielki, z waleczności znany. Mąż, wojownik niewielkiego wzrostu był a muskularny i barczysty. Ciężka broń protonowa u pasa mu dyndała, misiurka dendrytowa szyję zdobiła. Buty elektromagnetyczne trzymały go przy powierzchni, a i przykopać czasem pozwalały.

Samice ci mieli dorodne, choć jednako niewielkie. Trzymali je w siedzibach, co by po świecie nie wałęsały się, ferment siejąc.

Tysiączków kilka owych Extonów zamieszkiwało niewielką planetkę, której nazwa nic nikomu nie rzeknie. Dość, że żyli ci dostatnio w miarę, na krzyżowe wojny dawali werbować się, uprawiali smaczne śrubsztony i smarniki w swych ogródkach.

Aż razu pewnego jak grom z jasnego nieba uderzyły na nich wojska obce. Nieznane postacie sporych, błyszczących niklem wojów, co przybyły podbić spokojny lud. Bronił się on zaciekle, cały kunszt wkładając w obronę własnej ziemi. Na nic to. Masy najeźdźców zalały kraj, wybijając niewielkie zastępy tubylców.

I tu zaczęła się smutna historia podbitego ludu. Namiestnik Maniana gnębił ich bezlitośnie, na pożarcie Iwonom a synchronom rzucając. Podatkami ich gniótł, paszportów nie dawał, co dorodniejsze niewiasty do siebie kazał przyprowadzać.

Owa tyrania nie wpływała dobrze na jego popularność. Zawiązał się spisek, by obalić uprzykrzonego najeźdźcę. Ale jak to uczynić, skoro on chroniony jest polem, a i strażników oddanych ma wielu?

Dowiedzieli się spiskowcy, że niedaleko Maniana zbudować kazał planet kilka, skąd energię czerpał i gdzie obwody bezpieczeństwa miał swoje. Tak i postanowili zniszczyć centrum planet owych, co by tyrana się pozbyć.

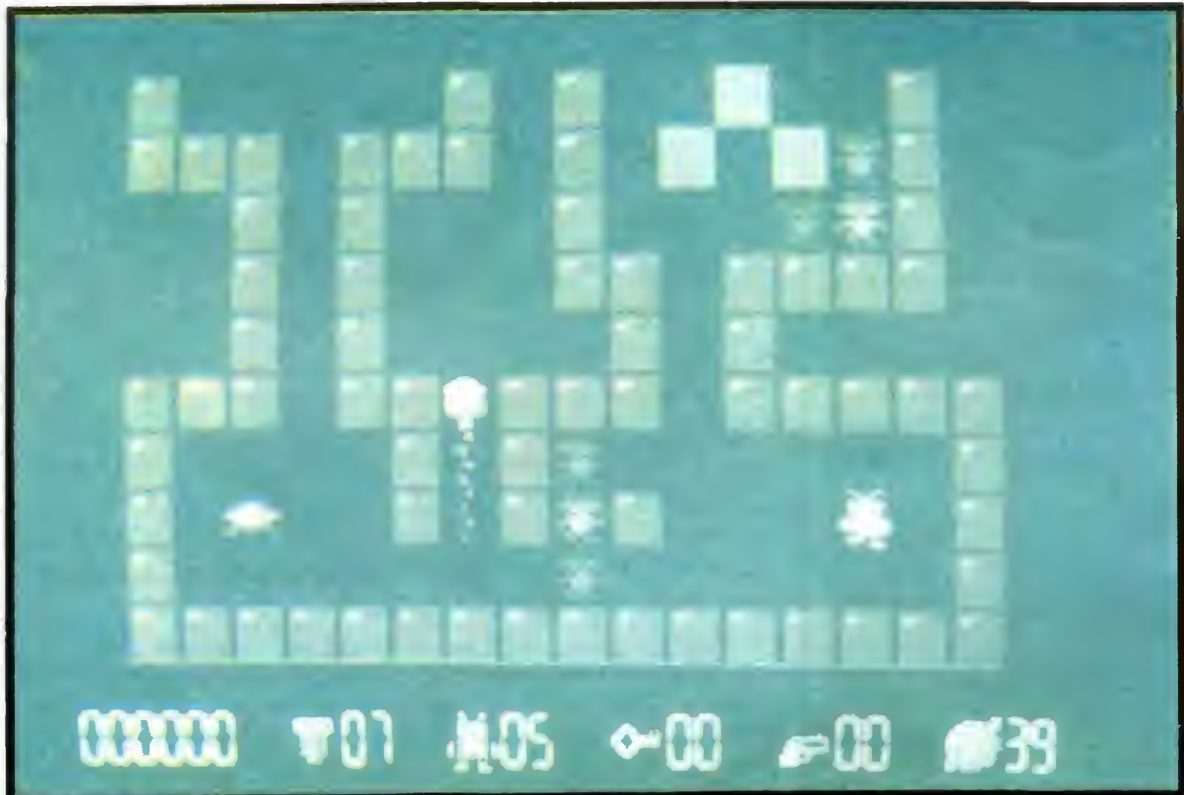
Całe trzy lata kradli i kupowali co się dało, by myśliwiec gwiezdny zbudować. Budową zajął się mistrz Electrox, co ze zmyślności w dziedzinie tej słynął. Myśliwiec ów, ze względów różnistych, miał być typu R. To znaczy, napęd jego fotonowy i odwrotnie skrętem skierowany.

W największej tajemnicy zmontowali myśliwiec i do ataku go przysposobili. Wybrali najzręczniejszego w statków pilotowaniu i on za sterami zasiadł. Zniszczyć musi system planet energetycznych docierając do ich jądra. Po drodze wiele niebezpieczeństw na niego czyha, ale i przyjazne konstrukcje spotka. Jego misja tajną jest i bardzo niebezpieczną. Jest nudna nieco również, ale cóż uczynić.

Co było po drodze i na miejscu, sami się przekonacie. A gdy wrócił bohater z misji swej dziejowej, na wolnej ziemi witali już go rodacy. A potem żyli długo i szczęśliwie, chociaż właściwie nie wiadomo.

Firma: Ocean
Komputer: Spectrum, Commodore, Amstrad, Atari ST, Amiga

Gen



ROBBO

Wstrętni Obcy porwali Robbo i umieścili go na jednej z planet odległego systemu gwiezdnego. Może on uwolnić się tylko przelatując małymi kapsułami ratunkowymi kolejno z jednej planety na drugą. Byłoby to bardzo proste — pomimo że planet tych jest aż 56 — lecz żadna z kapsuł nie jest w pełni sprawna. W celu ich uruchomienia Robbo musi pozbierać porozrzucane na planecie śrubki.

Ponieważ jest to wrogi system planetarny, to na każdej planecie czeka na naszego bohatera wiele niespodzianek. Większość z nich można rozpoznać po wyglądzie, choć przy niektórych długo musiałem się zastanowić — dopóki nie zadziałały. Są tam więc drzwi i klucze do nich, bomby, rozregulowane działa automatyczne, miotacze płomieni, amunicja do pistoletu, teleportery, magnesy, nietoperze, „diabełki", „oczy" i inne ciekawostki. Trzy ostatnie z wymienionych to nieprzyjemne stworki, które zabijają Robbo, jeśli się do nich zbliży. Najgorsze są „oczy", bo polują na Robbo, podczas gdy inne stworki nie zwracają na niego żadnej uwagi. Część z przedmiotów powinna być zbierana, gdyż będą one potrzebne do opuszczenia planety. Informacje o aktualnym stanie posiadania Robbo wyświetlane są w dolnej części ekranu.

Rozpoczynając grę, Robbo posiada osiem „żyć" i pistolet bez amunicji. Zarówno amunicję, jak i dodatkowe „życia" można znaleźć na niektórych planetach. Niestety, o ile dodatkowe „życie" pozostaje, to amunicja nie mieści się do kapsuły ratowniczej i na kolejnej planecie trzeba jej szukać od początku. Czasem napotyka się niespodziankę ukrytą w znaku zapytania. Należy w nią strzelić lub przesunąć tak, aby trafiło w nią działo. Konieczna jest jednak najwyższa ostrożność. Można bowiem w ten sposób zdobyć dodatkowe „życie", śrubkę, ale także mogą to być „oczy", które natychmiast rozpoczynają polowanie na Robbo.

Zabawa jest doskonała i gra wciąga niesamowicie. Nie trzeba tu wielkiej zręczności, choć trochę jej jest niezbędne. Przede wszystkim konieczne jest wykrycie sposobu wydostania się z każdej planety. Największą atrakcyjność gry stanowi niepowtarzalność rozwiązań — na każdej planecie inny jest sposób postępowania. Wprawdzie pewne elementy przypominają inne gry, lecz ich połączenie jest oryginalne i dało doskonały efekt.

Obsługa gry jest bardzo prosta, a ponadto po wczytaniu gry wyświetlana jest instrukcja w języku polskim. Robbo sterowany jest joystickiem lub klawiszami ze strzałkami. Należy jednak bardzo uważać — niekiedy błędne wykonanie kilku pierwszych ruchów uniemożliwia już opuszczenie planety. Nie powinno się także wykonywać zbędnych przesunięć obiektów (tych, które można przesuwać), gdyż pozornie przeszkadzające skrzynki, bomby itp. mogą się w końcu okazać niezbędne do przejścia na następny poziom.

Na gracza, który szczęśliwie pokona wszystkie poziomy i zakończy grę, czeka niespodzianka: na ekranie pojawi się napis. Producent obiecuje pierwszym pięciu graczom, którzy podadzą treść tego napisu, bezpłatny egzemplarz następnej gry o roboczym tytule „Misja".

Na zakończenie informacja dla tych, którym nie wystarcza osiem „żyć". Podczas wyświetlania instrukcji po odczytaniu programu należy wpisać na klawiaturze imię i nazwisko autora oddzielone od siebie spacją:

„JANUSZ PELC"

Po naciśnięciu przycisku joysticka gra rozpoczyna się normalnie, lecz teraz Robbo ma 80 „żyć"!

(ziew)

Autor: Janusz Pelc
Producent: Laboratorium Komputerowe „Avalon"
Komputer: Atari XL/XE

ZAGRAJMY W TO JESZCZE RAZ

```
FAIRLIGHT                  34413,234
                           34414,234
                           34420,234
                           34421,234
                      SYS   20992 L
FALCON PATROL              16764,234
                           16765,234
                      RUN
FALCON PATROL              16764,36
                           16705,2
                      SYS   16640
FALCON PATROL 2             9564,234
                            9596,234
FEUD                       16404,15
                           17591,1-40
                      SYS   16384
FINAL CONQUEST             23407,234
                           23408,234
                           23409,234
                           23415,234
                           23416,234
                           23417,234
                           23423,234
                           23424,234
                           23425,234
FINDERS KEEPERS            29787,76
                           29788,96
                           29789,116
                      SYS   49152 L
FIRE ANT                   17568,100 L
                          *17568,20  L
FIRE FLY                    4527,165
                          * 4301,173
                      SYS    4301
FIRE TRAP                   7500,234
                            7501,234
                      SYS    4096 L
FIRELORD                    7852,205 L
                          * 3740,99
                            5721,238
                      SYS    2304
FIRETRACK                  12285,234
                           12286,234
                           12287,234
                      SYS    9216 L
FIRETRAP                    7500,234
                            7501,234
                      SYS    4096
FIST 2                      6701,255
FLAK                        4798,36  L
FLASCHBIER 2                9963,205
                      SYS   26763 L
FLASH GORDON               25903,234

                           25904,234
                      SYS   12288 L
FLIP & FLOP                16505,0   L
FLYING SHARK               54462,201 L
FRIGHTMARE                 21840,0
                           21841,4
                      SYS   16384 L
FROGGER                    22341,173
                           22347,173 L
FROGGER SEGA               22341,173 L
FRANTIC FREDDIE            34535,24  L
                          * 3118,234
                            3119,234
                          *36364,234 L
                          *31887,255   ?
                           31887,50    ?
FORT APOKALYPSE            36364,234
                           36365,234
                           36366,234
                           36367,234 L
                          *36339,153 L
                          *36539,255 L
                          *36353,234 L
                          *36366,153 L
                          *14697,0   BONUS
                           14760,0   TANK
                           36366,0
                          *31887,50
                          *34535,24


G

GALAGA                     17388,165 L
                          *17288,165
                          *17383,230
                          *17788,165
                           17383,173
                           17388,173
GALAXIONS-ACTIVI.           7865,230
GALAXIONS                   7065,230
                           17288,165 L
                          * 7365,230   ?
```

REWELACJA ROKU: SOUND TRACKER!

Każdy komputer jest wart tylko tyle, ile programy do niego. To tłumaczy ekspansję IBM-ów oraz niebywałą popularność Spectrum. Ciągle pojawiające się gry, użytki i „demka" nie pozwalają wielu użytkownikom pozbyć się swoich Spectrumienek.

Stacja dysków i generator AY to teraz podstawowe pudełka na stole z komputerkiem. Stacja oprogramowana jest słusznie, zważywszy niekonsekwencje portugalskiego producenta, który nie dość, że wyprodukował urządzenie bez jakiejkolwiek dokumentacji, to całkowicie umył ręce od produkcji oprogramowania. Na szczęście Polak potrafi i mamy już mnóstwo użytków i przeróbek gier. Oprócz tego nad zgłębieniem tajemnic systemu TOS pracują nasi najlepsi i rezultaty poznacie wkrótce.

Natomiast AY-grek oprogramowany jest wprost oszałamiająco. Niezliczona ilość „dem"-onstracji i gier mile dla ucha wykorzystuje to cudo. Możliwe jest również tworzenie muzyki, a to za pośrednictwem opisywanego niedawno programu **The Music Box 128**. Jest on niezły, lecz z dzisiejszej perspektywy jedynie zadowalający.

Powstało jednak COŚ, co odwróci głowy muzyków i muzykantów z powrotem w kierunku Spectrum. Twór nazywa się **Sound Tracker** i jest **REWELACYJNY!**

Jak to było? Twórczo zapłodniony przez p. Macieja Wrońskiego — z zawodu muzyka, z zamiłowania Spectrumowca — znany posiadaczom So-So Copy, Mega Phantoma i graczom Jarosław Burczyński pół roku pracował nad prawdziwym programem muzycznym. Inwencje i niektóre rozwiązania czerpał z „playerów", czyli odgrywaczy oryginalnych melodii z gier, zaś szatę graficzną i ogólną konstrukcję zapożyczył z wersji z Amigi, ulepszając je nieco.

Każdemu, kto ujrzał **Sound Trackera** w akcji, przychodzi do głowy pięć liter: AMIGA. My byliśmy tymi szczęśliwcami, którzy mieli okazję testować program, zanim ktokolwiek o nim usłyszał. I gwarantujemy — jest taki, jaki powinien być oryginalny zachodni program użytkowy, sprzedawany w domach towarowych za $99.99.

Główna myśl to zapis muzyki w postaci trzech słupków nut. Nuty opisywane są oznaczeniem literowym (klawiatura jak w fortepianie); dostępne jest osiem oktaw. Słupki mają do 64 wierszy i stanowią tzw. patterny, czyli tablice danych dla trzech kanałów. Pamięć pozwala na zapisanie do 31 takich tablic, przy czym ich długość może być mniejsza.

Oprócz oznaczeń nut i numerów oktaw w patternach zapisuje się (oddzielnie dla każdego dźwięku): numer brzmienia (instrumentu) i komendę ustalającą standardową obwiednię AY-greka lub ozdobnik.

Wbudowanym edytorem można zdefiniować 15 brzmień, podając w postaci diagramu układ głośności i szumów oraz rozkład półdźwięków wchodzących w zakres brzmienia. Każde brzmienie może być oprócz tego zmodyfikowane ozdobnikiem, więc teoretycznie do dyspozycji są 1024 brzmienia.

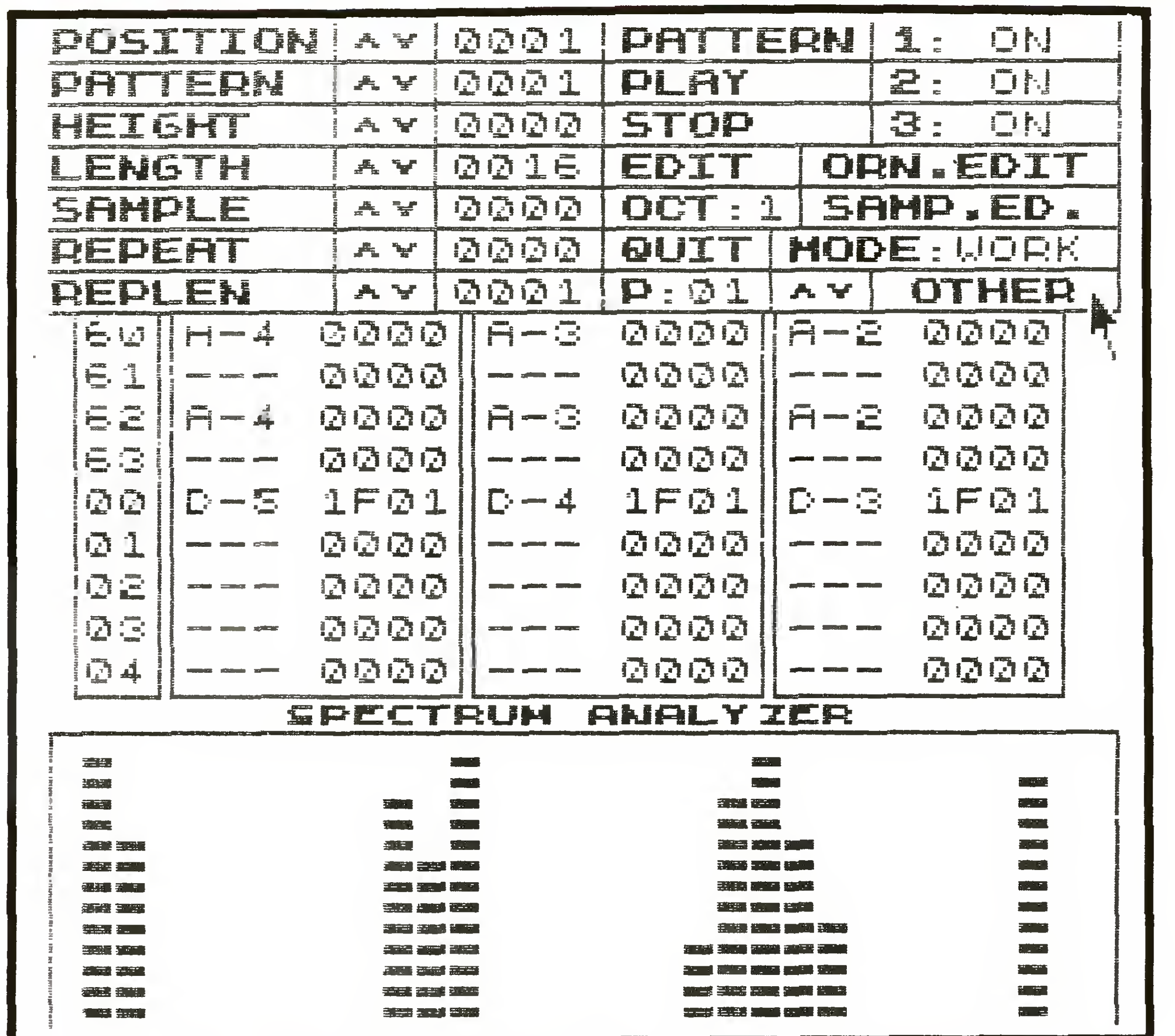
Program używa dwóch menu: edycyjnego i pozostałych opcji. Jako pamięć zewnętrzna pomyślany został dysk, lecz w toku są prace nad wersją taśmowo-dyskową.

Możliwy jest zapis i odczyt: ułożonego utworu, jednego z 15 brzmień i jednego z 15 ozdobników. Oprócz tego zmiana tempa, przenoszenie i czyszczenie fragmentów utworu bądź transpozycja nut.

Menu edycyjne pozwala na zapisanie patternów, zdefiniowanie brzmień i ozdobników oraz ułożenie utworu. Ta ostatnia czynność polega na podaniu kolejności patternów, które mają być odgrywane. Bardzo ciekawy jest system ustalania tej kolejności, przez przewijanie dwóch liczników — „Position" i „Pattern". Podobny system posiadają komendy zapętlania obwiedni dla brzmień.

Pełny wykaz możliwości i szczegółowy opis opcji znajduje się w obszernej instrukcji dołączanej do programu.

Autor nie zapomniał i o bajerach, jak np. trzydziestosłupkowy kolorowy analizator częstotliwości dźwięku, porządna oprawa graficzna, strzałka sterowana joystickiem i klawiatur oraz możliwość włączania i wyłączania kanałów (podczas gry).



Napisaną muzykę, która w wersji źródłowej może zajmować do 20 KB, można skompilować osobnym programem. Jest on zadziwiający, gdyż oprócz tego, że odgrywa muzykę podczas współpracy ze stacją i kompilacji, to tworzy z tekstu źródłowego muzyki kod wynikowy z playerem, do użycia we własnej (lub cudzej) grze, demie itp. Kod generowany jest od dowolnego adresu i odpowiada standardom w swej dziedzinie.

Na **Sound Trackerze** powstało już kilka melodii autorstwa w.wym. p. Wrońskiego. Można się spodziewać wysypu wspaniałych dem, jak tylko upowszechni się **Sound Tracker**. Jestem pewien, że w końcu trafi i za granicę, gdzie wielu ludziom opadną szczęki.

Tymczasem autor, Jarosław Burczyński, ma lat 18. Jest zadziwiające, kto naprawdę potrafi w Polsce pisać programy. Przeważająca większość współpracowników „Bajtka" to ludzie w tym wieku lub młodsi. To do nich należy przyszłość informatyki, jeśli pójdą w dobrym kierunku. Dziś są pionierami, samoukami, jutro będą kształcić wysoko kwalifikowane kadry programistów. Okazuje się, że nie nowoczesny sprzęt i nie najnowsze oprogramowanie tworzy fachowca, lecz prawdziwa pasja.

Marcin Przasnyski

I·N·L·I·N·E

Pisanie programów w Pascalu na Spectrum jest bardzo przyjemne do momentu, w którym chcemy skorzystać z jakiegoś niestandardowego rozwiązania — na przykład z podprogramu w kodzie maszynowym. Jeżeli nawet pominiemy wszystkie problemy związane z przekazywaniem danych, zostanie nam jeszcze jeden kłopot — mozolne przepisywanie kodu maszynowego do instrukcji **inline**. Nie ma problemu, gdy kawałek kodu ma pięć czy dziesięć bajtów, chociaż i to potrafi doprowadzić człowieka do białej gorączki, zwłaszcza gdy równocześnie trzeba dokonywać „ręcznej" asemblacji. Gorzej, gdy podprogram składa się z kilkudziesięciu albo i więcej bajtów. Żeby uniknąć kłopotów, przygotowałem prezentowany program. Przy jego pomocy można nagrać na taśmę gotową instrukcję **inline**, w formacie stosowanym przez kompilator. Przed uruchomieniem programu trzeba w pamięci komputera przygotować, np. przy pomocy GENSA, kod przeznaczony do instrukcji **inline**. Program zapyta o trzy rzeczy — adres początkowy i końcowy kodu oraz o adres pierwszego bajtu obszaru, w którym zostanie przygotowany blok danych do nagrania na taśmę. Po przygotowaniu danych potrzebna jeszcze będzie nazwa i można nagrywać (i weryfikować).

Marcin Borkowski

```
 20 INPUT "Początek kodu ";ad1
 30 INPUT "Koniec kodu ";ad2
 40 INPUT "Pole robocze ";robo
 50 LET rob=robo: LET nl=1: GO SUB 500
 60 POKE rob,1: LET rob=rob+1
 70 LET h$="INLINE(": GO SUB 400
 80 LET licz=0
 90 LET h$=STR$ PEEK ad1+","
100 GO SUB 400
110 LET licz=licz+1: LET ad1=ad1+1
120 IF ad1=ad2 THEN GO TO 170
130 IF licz<>6 THEN GO TO 90
140 LET h$=CHR$ 13: GO SUB 400
150 LET nl=nl+1: GO SUB 500
160 LET h$=CHR$ 8: GO SUB 400: GO TO 80
170 LET rob=rob-1: LET h$=");"+CHR$ 13
180 GO SUB 400
190 INPUT "Nazwa zbioru "; LINE n$
200 LET nl=rob-robo: GO SUB 500
210 SAVE n$CODE rob-2,2: SAVE n$CODE robo,nl
220 PRINT "Przewiń taśmę do weryfikacji..."
230 VERIFY n$CODE : VERIFY n$CODE
240 STOP
400 FOR i=1 TO LEN h$
410 POKE rob,CODE h$(i): LET rob=rob+1
420 NEXT i: RETURN
500 POKE rob,nl-256*INT (nl/256)
510 POKE rob+1,INT (nl/256)
520 LET rob=rob+2: RETURN
```

SYNTHSET — cz. 1

Program SYNTHSET służy do cyfrowego zapamiętywania dźwięków w pamięci komputera — czyli do tworzenia SAMPLINGU. Na czym polega istota owego procesu zwanego samplingiem (czy jak kto woli, próbkowaniem)? Jest to proste: komputer co pewien czas testuje port, do którego jest podłączone urządzenie wytwarzające dźwięk i odpowiednio zapisuje w pamięci jedynkę lub zero (bity).

Od długości przerw między takimi testami uzależniona jest jakość samplingu — im te przerwy są krótsze, tym dźwięk zostanie wierniej zapisany w pamięci maszyny, ale zajmuje więcej miejsca.

Mój program testuje port gniazda **EAR (IN #FE)**, a odtwarza zapisane dźwięki za pomocą układu **AY3-8912/10**. Jest to zatem narzędzie do tworzenia samplingu na popularny interfejs **SOUND 128**.

Aby otrzymać poprawną wersję programu, należy dokładnie wpisać listing 1 i ochronić go na kasetę (GOD SAVE THE PROGRAM) zleceniem **SAVE "SYNTHSET" LINE 1.**

Drugą czynnością jest wpisanie listingu 2. Gdy już to zrobimy, trzeba wykonać komendę **RUN** i poczekać kilka sekund, aż program wpisze wartości do odpowiednich komórek pamięci i wypisze komunikat o zamiarze nagrania bloku kodu na kasetę (program poinformuje nas o ewentualnych błędach w liniach DATA).

Przychylając się do prośby w oknie systemowym — nagramy kod i otrzymamy pełną wersję programu SYNTHSET.

Po wyzerowaniu komputera cofamy taśmę i wpisujemy **LOAD "SYNTHSET".**

Gdy program zostanie załadowany do pamięci — uruchomi się automatycznie i naszym zmęczonym oczom ukaże się menu z opcjami oraz kursor w postaci strzałki kierowanej systemem Kempston.

Poruszając joystickiem (tu: dla polonistów Dżojstyk), co powoduje ruch kursora, najeżdżamy na linię z opcją i naciskamy przycisk strzału. Opcja zostanie zainicjowana.

Przy niektórych liniach z opcjami są numerki i oznaczają one aktualną wartość danej funkcji opisywanej odpowiednią opcją.

Program posiada 9 opcji, oto one:

RECORD — opcja wczytywania dźwięków do komputera łącząc urządzenie dźwiękotwórcze z gniazdem **EAR** i uruchamiając tę opcję musimy ją potwierdzić przez „y" (program pyta **SURE (Y/N),** pojawi się wtedy napis **ENTER:** i program czeka na wciśnięcie tego klawisza — bezpośrednio po jego naciśnięciu zostanie rozpoczęte próbkowanie.

PLAY — opcja powoduje odtworzenie przez układ **AY3-8912/10** zapisanych w pamięci danych. Należy oczywiście podłączyć go do wzmacniacza, tak jak przy słuchaniu melodyjek z programów demonstracyjnych.

LENGTH — ustawianie długości wczytywanych dźwięków a właściwie długości obszaru pamięci komputera, jaki te dźwięki będą zajmować. Można wybrać parametr od 1 do 80, gdzie jedna jednostka oznacza 256 bajtów pamięci, bądź przy standardowo ustawionych opóźnieniach czas ok. 0.28 sek.

INVERSION — powoduje odwrócenie kolejności bajtów będących danymi dźwięku, tzn. bajt pierwszy staje się ostatnim, a ostatni staje się pierwszym. Pozwala to na osiągnięcie bardzo ciekawych efektów dźwiękowych.

PLAY DELAY — jest to opcja pozwalająca na zmianę szybkości odtwarzania dźwięku. Po jej zainicjowaniu program czeka na wprowadzenie opóźnienia, z jakim będzie odtwarzał dane z pamięci, może to być liczba całkowita z przedziału od 3 do 30 (standardowo: 12).

RECORD DELAY — pozwala na ustawienie częstotliwości próbkowania dźwięku z gniazda **EAR**. Standardowo opóźnienie to wynosi 15, lecz my możemy ustawić je na dowolną wartość z przedziału (3,30). Im opóźnienie to jest mniejsze, tym jakość dźwięku większa — w ten sposób skracamy czas zapisywania danych w pamięci komputera, co z kolei powoduje większą „pamięciożerność".

VOLUME — opcja służy do ustawiania głośności odtwarzania dźwięku przez układ AY3-8912/10. Zakres, jakim możemy operować, to 1–15, standardowo wartość tej opcji wynosi 10.

DIAGRAM — bezpośrednio po wywołaniu tej opcji w dolnej części ekranu ukaże się odwzorowanie graficzne obszaru pamięci o długości wskazywanej przez opcję **LENGTH**. Gdy w pamięci będą rezydowały dane, wykres przez swą uniwersalność jest pomocny w ich obróbce.

SAVE DATA — gdy uznamy, iż dźwięki, które zapisaliśmy w pamięci, są warte utrwalenia, używamy tej opcji do zgrania danych na taśmę magnetofonową. Najpierw podajemy nazwę, jaką chcemy nadać naszemu plikowi, w odpowiedzi na pytanie **BEGIN** (16384-...) podajemy adres, pod jakim będziemy używać naszych danych (jest to adres ładowania w nagłówku pliku), aż w końcu uruchamiamy magnetofon i naciskamy dowolny klawisz. Dane będą zapisywane w formie pliku typu „bytes" o wcześniej podanym adresie ładowania.

Wzajemne stosowanie funkcji **LENGTH** i **INVERSION** pozwala na wycinanie i mieszanie fragmentów danych. Jest to pomocne w wybieraniu interesujących nas części danych.

Program SYNTHSET służy do tworzenia i obróbki danych, których możemy użyć we własnym programie. Potrzebny będzie do tego „player", który odtworzy dane.

Procedurę odtwarzającą dane i inne niuanse związane z samplingiem przedstawię w kolejnym odcinku.

Maciej Brømba Pietraś

```
  5 REM SYNTHSET by BROMBA'89
 10 BORDER VAL "7": INK NOT PI: PAPER VAL "7": CLEAR VAL "32767": LOAD "SYNTHSE
T.1"CODE : CLS
 15 LET lem=80: LET inwe=0: LET dzap=15: LET dod=12: LET vol=10
 16 PRINT  PAPER 7; BRIGHT 1; INK 2;AT 2,21;"~SYNTHSET~";AT 3,21;"  BROMBA  ";A
T 4,21;"  1989  "
 20 PLOT 166,161: DRAW 83,0: DRAW 0,-27: DRAW -83,0: DRAW 0,27
 30 PRINT  BRIGHT 1; PAPER 2; INK 7;AT 1,1;" RECORD        ";AT 3,1;" PLAY
  ";AT 5,1; BRIGHT 0;" LENGTH        ";AT 7,1;" INVERSION    ";AT 9,1; PAPER 3;"
 PLAY DELAY   ";AT 11,1;" RECORD DELAY ";AT 13,1;" VOLUME        ";AT 15,1; INK 6
;" DIAGRAM       ";AT 17,1; PAPER 4; INK 1;" SAVE DATA    "
 35 FOR f=0 TO 18 STEP 2: PRINT AT f,1; PAPER 6;"               ": NEXT f
 36 PLOT 7,24: DRAW 113,0: DRAW 0,151: DRAW -113,0: DRAW 0,-151
 37 IF inwe=0 THEN  PRINT AT 7,16;"->"
 38 IF inwe=1 THEN  PRINT AT 7,16;"<-"
 39 PRINT AT 5,16;lem;"  ";AT 9,16;dod;"  ";AT 11,16;dzap;"  ";AT 13,16;vol;" "

 41
 42 LET len=256*lem: RANDOMIZE len: LET len1=PEEK 23670: LET len2=PEEK 23671
 45 POKE 60005,len1: POKE 60006,len2: POKE 60097,len1: POKE 60098,len2: POKE 60
241,len1: POKE 60242,len2: POKE 60261,len1: POKE 60262,len2
 50 LET lena=(len/2): RANDOMIZE lena: POKE 60199,PEEK 23670: POKE 60200,PEEK 23
671
 60 LET end=32768+len: RANDOMIZE end: POKE 60196,PEEK 23670: POKE 60197,PEEK 23
671
 65 POKE 60010,dzap: POKE 60043,(dzap-2): POKE 60102,dod: POKE 60129,(dod-2)
 70 POKE 60061,vol
 99 PRINT AT 16,20;"          "
100 LET kur=USR 60268
110 IF kur>=257 AND kur<=270 THEN  GO TO 700
120 IF kur>=769 AND kur<=782 THEN  GO TO 600
125 IF kur>=1281 AND kur<=1294 THEN  GO TO 500
130 IF kur>=1793 AND kur<=1806 THEN  GO TO 450
135 IF kur>=2305 AND kur<=2318 THEN  GO TO 400
140 IF kur>=2817 AND kur<=2830 THEN  GO TO 300
145 IF kur>=3841 AND kur<=3854 THEN  GO TO 170
150 IF kur>=4353 AND kur<=4366 THEN  GO TO 200
152 IF kur>=3329 AND kur<=3342 THEN  GO TO 350
155 GO TO 100
170 FOR f=19 TO 21: PRINT AT f,0;"                                ": NEXT f: IN
K 4: LET pl=0: FOR w=32768 TO (32768+len) STEP 81
175 PLOT pl,0: DRAW 0,((PEEK w)/10)-5*(((PEEK w)/10)>5)
180 LET pl=pl+1
190 NEXT w
195 BEEP .4,-4: INK 0: GO TO 100
200 INPUT "NAME: "; LINE n$: IF n$="" OR LEN n$>10 THEN  GO TO 200
210 FOR f=60251 TO 60260: POKE f,32: NEXT f
215 LET a=1: FOR f=60251 TO (60251+LEN n$-1): POKE f,CODE n$(a): LET a=a+1: NEX
T f
220 INPUT "BEGIN (16384-";(65535-len);"): ";styr: IF styr<16384 OR styr>(65535-
len) THEN  GO TO 220
230 RANDOMIZE styr: POKE 60263,PEEK 23670: POKE 60264,PEEK 23671
240 PRINT #0;"start tape": PAUSE 0: INPUT ': RANDOMIZE USR 60219: PAUSE 50: BEE
P .4,-4: GO TO 37
300 INPUT "RECORD DELAY (3-30): ";a: IF a<3 OR a>30 OR a<>INT a THEN  GO TO 300

310 LET dzap=a: GO TO 37
350 INPUT "VOLUME (1-15): ";a: IF a<1 OR a>15 OR a<>INT a THEN  GO TO 350
360 LET vol=a: GO TO 37
400 INPUT "PLAY DELAY (3-30): ";a: IF a<3 OR a>30 OR a<>INT a THEN  GO TO 400
410 LET dod=a: GO TO 37
450 LET inwe=NOT inwe
460 RANDOMIZE USR 60191
470 BEEP .2,10: BEEP .1,-5
475 GO TO 37
500 INPUT "LENGTH (1-80): ";a: IF a<1 OR a>80 OR a<>INT a THEN  GO TO 500
510 LET lem=a: GO TO 37
600 PRINT AT 16,23; INVERSE 1;" PLAY "
610 RANDOMIZE USR 60050: GO TO 99
700 INPUT "SURE (Y/N) ? "; LINE b$: IF b$<>"Y" AND b$<>"y" THEN  GO TO 37
710 PRINT AT 16,22; INVERSE 1; PAPER 4;" ENTER: "
715 PAUSE 0: LET inwe=0
720 IF CODE INKEY$<>13 THEN  GO TO 715
730 PRINT AT 16,20; PAPER 6; FLASH 1; BRIGHT 1;" RECORDING: "
740 RANDOMIZE USR 6e4: GO TO 37
750
800 REM NIETZSCHE OK
```

LISTING 1

LISTING 2

```
   1 DATA 243,33,0,128,17,0,0,6,8,14,449
   2 DATA 15,13,32,253,219,254,203,119,32,7,1147
   3 DATA 8,55,55,23,8,24,7,8,55,63,306
   4 DATA 23,8,62,0,16,229,8,119,35,27,527
   5 DATA 6,8,14,13,122,179,32,219,251,201,1045
   6 DATA 1,253,255,62,9,237,121,1,253,191,1383
   7 DATA 62,14,237,121,1,253,255,62,3,237,1245
   8 DATA 121,1,253,191,62,0,237,121,1,253,1240
   9 DATA 255,62,2,237,121,1,253,191,62,0,1184
  10 DATA 237,121,243,33,0,128,17,0,0,6,785
  11 DATA 8,14,15,13,32,253,203,6,56,7,607
  12 DATA 197,205,232,234,193,24,5,197,205,247,1739
  13 DATA 234,193,16,233,35,27,6,8,14,13,779
  14 DATA 122,179,32,225,251,201,1,253,255,62,1581
  15 DATA 7,237,121,1,253,191,62,253,237,121,1483
  16 DATA 201,1,253,255,62,7,237,121,1,253,1391
  17 DATA 191,62,255,237,121,201,0,0,0,0,1067
  18 DATA 0,0,0,0,0,0,0,0,0,0,0
  19 DATA 0,0,0,0,0,0,0,0,0,0,0
  20 DATA 0,0,33,0,128,17,0,0,1,0,179
  21 DATA 0,197,70,26,72,71,121,112,18,35,722
  22 DATA 27,193,11,120,177,32,240,201,0,221,1222
  23 DATA 33,90,235,17,17,0,62,0,55,205,714
  24 DATA 194,4,6,255,16,254,221,33,0,128,1111
  25 DATA 17,0,0,62,255,55,205,194,4,201,993
  26 DATA 3,49,50,51,52,53,54,55,56,57,480
  27 DATA 48,0,0,0,0,197,7,0,195,121,568
  28 DATA 236,127,95,1,0,0,0,63,64,31,617
  29 DATA 96,15,112,7,120,3,124,1,126,0,604
  30 DATA 120,1,72,3,8,163,4,241,4,241,857
  31 DATA 2,248,2,248,0,252,0,0,0,0,752
  32 DATA 0,0,0,0,0,0,0,0,0,0,0
  33 DATA 0,0,0,0,0,0,0,0,0,0,0
  34 DATA 0,0,0,0,0,0,0,0,0,0,0
  35 DATA 0,0,0,0,0,0,0,0,0,0,0
  36 DATA 0,0,0,0,0,0,0,0,66,82,148
  37 DATA 48,77,66,65,0,237,75,111,235,237,1151
  38 DATA 67,114,235,253,33,146,235,33,116,235,1467
  39 DATA 30,15,205,52,236,24,3,205,45,236,1051
  40 DATA 216,213,197,221,126,0,253,119,0,253,1598
  41 DATA 35,221,126,1,253,119,0,253,35,66,1109
  42 DATA 86,30,0,35,126,14,255,35,4,5,590
  43 DATA 40,11,55,31,203,25,167,203,26,203,964
  44 DATA 27,16,245,221,166,0,170,221,119,0,1185
  45 DATA 221,125,246,224,60,40,8,221,126,1,1272
  46 DATA 161,171,221,119,1,193,209,29,32,183,1319
  47 DATA 201,221,36,4,120,230,7,192,62,191,1264
  48 DATA 184,216,229,120,205,178,34,87,229,221,1703
  49 DATA 225,225,201,6,0,33,111,235,175,219,1430
  50 DATA 21,203,79,40,1,5,203,71,40,1,664
  51 DATA 4,72,245,126,128,254,255,40,1,119,1244
  52 DATA 241,6,0,203,95,40,1,5,203,87,881
  53 DATA 40,1,4,35,126,128,254,191,48,1,828
  54 DATA 119,120,177,35,119,201,0,243,205,205,1424
  55 DATA 235,58,113,235,31,48,6,205,161,236,1328
  56 DATA 205,205,235,205,67,236,1,220,5,11,1390
  57 DATA 120,177,32,251,175,219,21,254,0,40,1289
  58 DATA 250,203,103,32,34,24,220,33,146,235,1280
  59 DATA 237,75,114,235,30,15,205,52,236,24,1223
  60 DATA 3,205,45,236,216,126,221,119,0,35,1206
  61 DATA 126,221,119,1,35,29,32,239,201,205,1208
  62 DATA 161,236,42,111,235,203,61,203,61,203,1516
  63 DATA 61,77,203,60,203,60,203,60,68,251,1246
  64 DATA 201,0,0,0,0,0,0,0,0,0,201
8000 CLEAR 5e4: LET a=60000
8010 FOR n=1 TO 64
8020 LET s=0
8030 FOR m=0 TO 9
8040 READ w: POKE a,w: LET s=s+w: LET a=a+1
8050 NEXT m: READ w: IF w<>s THEN  PRINT "Popraw
     linie ";n: STOP
8060 NEXT n
8070 SAVE "synthset.1"CODE 60000.635
```

JUNIOR W SZKOLE

Niewiele listów przychodzi do nas w związku z obecnością Juniorów w szkołach. Zgodnie z obietnicą drukujemy adresy szkół, które o to proszą. Oto kolejny adres:
Zespół Szkół Zawodowych nr 1 ul. Wojska Polskiego 66 63-200 Jarocin, tel. 26-19 posiadamy 10 Juniorów, w tym 2 autonomiczne, 3 drukarki D-100. Dysponujemy również dużą liczbą krótkich programików z różnych przedmiotów.

wolniej, wolniej...

Jednym z ciekawszych interface'ów do ZX SPECTRUM jest SLOMO. Jego zadaniem jest zwolnienie pracy procesora, a więc i wykonywanego programu.

Układ elektroniczny jest bardzo prosty zawiera tylko jeden układ scalony, potencjometr oraz 2 kondensatory. Połączenie z szyną komputera także jest łatwe do wykonania, interface wykorzystuje zaledwie 4 sygnały z szyny: CLK (zegar systemowy), BUSRQ („blokuje" pracę procesora) oraz napięcie zasilające 5V i masę.

Układ „zawiesza" procesor na pewien czas, określony przez potencjometr, a czasem odniesienia jest zegar systemowy CLK (3,5 MHz). Przystawka umożliwia płynne zwolnienie pracy programu, aż do całkowitego zatrzymania.

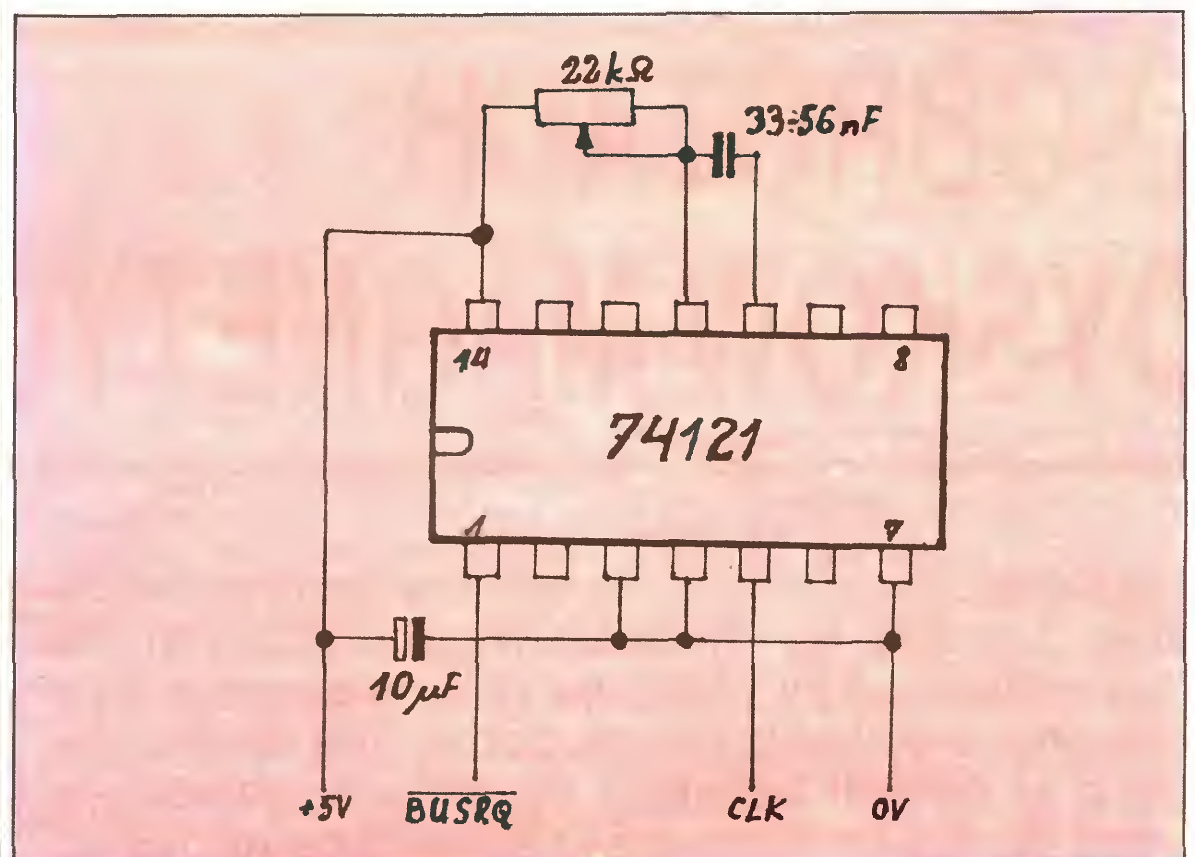
Jako US zastosowano układ TTL 74121 — polski odpowiednik UCY 74121. Nie musi on być w technologii TTL LS, bo używane przez niego sygnały nie są wykorzystywane przez inne interface-y. Całość najlepiej umieścić w innym „międzymordziu" np. Kempstonie (zmieści się, właściwie to tylko 1 scalak!). Odradzam robienie przystawki, pomimo jej prostoty, osobom mającym pierwszy kontakt z lutownicą.

W czasie uruchamiania należy potencjometr ustawić w skrajne położenie (przewody zwarte), gdyż inaczej start systemu będzie trwał długo.

Kondensator 33-56 nF należy dobrać doświadczalnie (im większa pojemność, tym większe zwolnienie). Z wykonanych przeze mnie 3 egzemplarzy wszystkie od razu pracowały poprawnie.

I na koniec UWAGA:
PODŁĄCZEŃ CZEGOKOLWIEK DO SZYNY KOMPUTERA MOŻNA DOKONYWAĆ JEDYNIE PRZY ODŁĄCZONYM ZASILANIU!!!

Grzegorz Ostapiuk



Spis elementów:
● potencjometr 22 kΩ ● kondensator 33÷56 nF ● kondensator 10µF (blokujący napięcie +5 V) ● układ scalony 74121 (polski zamiennik UCY 74121) ● złącze krawędziowe (jeśli montujemy SLOMO w innym interface złącze jest zbędne).

SPECTRUMOWA LISTA PRZEBOJÓW

Jak dowiadujemy się z korespondencji, nasza lista ma spore powodzenie. Dziś w kategorii użytków królują programy naszych autorów. Te i inne programy użytkowe dostępne są w Redakcji. Wśród nich m.in. **File Compressor** oszczędzający miejsce na dyskach i taśmach, **DCMR** katalogujący programy na dyskietkach, **COMALL** kopiujący na dysk więcej i lepiej niż **Zebra Copy.** Oprócz tego znany już plik kopierów dysk-dysk oraz rewelacyjny program do tworzenia muzyki — **SOUND TRACKER.**

A oto notowanie:

Grafika:

1. SAVAGE
2. ALTERED BEAST
3. HARD DRIVIN
4. UNTOUCHABLES
5. STRIDER

Muzyka 48:

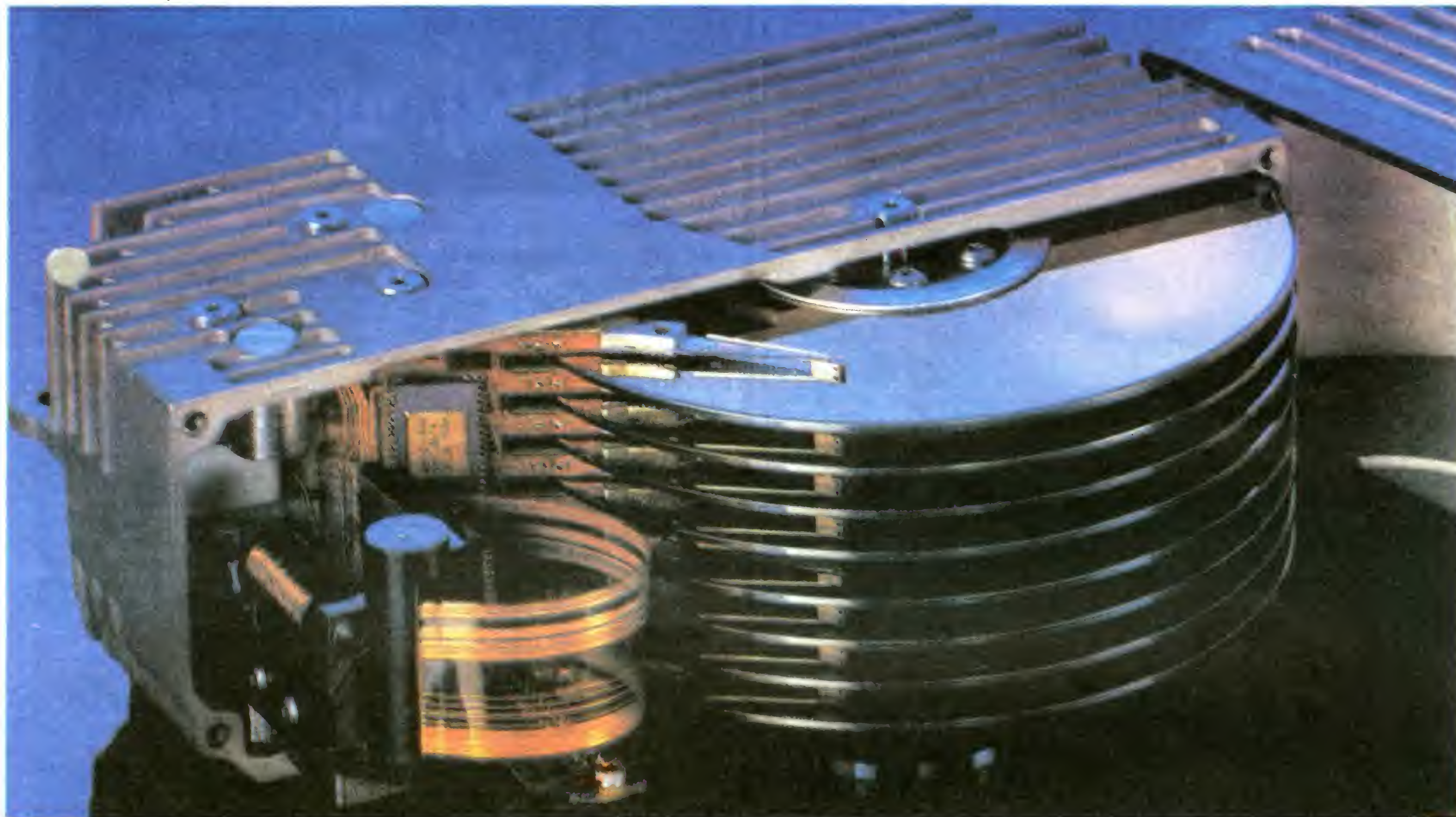
1. CHASE HQ
2. ZANTHRAX
3. TRANTOR
4. BATMAN
5. SAVAGE

Muzyka 128:

1. UNTOUCHABLES
2. MUNSTERS
3. KGB SUPERSPY
4. PLATOON
5. GHOSTBUSTERS II

Użytki:

1. FILE COMPRESSOR
2. TURBO PASCAL 3.0
3. DEERCOPY
4. PET/TEXT-ED
5. FOX MONITOR



O OBROTACH DYSKÓW MAGNETYCZNYCH cz. II

Ostatnio omówiliśmy w skrócie budowę i działanie pamięci na dyskach magnetycznych, na dziś zostawiliśmy przegląd tych własności dysków, które mają wpływ na codzienną eksploatację, gdy ścieżki i sektory fizyczne są zasłonięte przed użytkownikiem przez system operacyjny.

Nie znaczy to jednak, że wolno nam zupełnie zapomnieć o fizycznej naturze urządzeń. Jeśli nawet nie chcemy wnikać w szczegóły techniczne, to pamiętajmy o minimum niezbędnej dbałości i konserwacji. Aby uświadomić sobie wagę problemu, proponuję dokładnie obejrzeć poniższy rysunek, na którym przedstawiono w tej samej skali głowicę magnetyczną pracującą na powierzchni dyskietki oraz cząsteczkę dymu (myślący nie pali!), odcisk palca, cząsteczkę kurzu i włos. Myślę, że rysunek ten nie wymaga komentarza.

Jak już stwierdziliśmy, w codziennej pracy z dysków korzystamy za pośrednictwem systemu operacyjnego. Informacje przeznaczone do przechowania grupowane są w tzw. zbiory (ang. file). Chcąc np. utworzyć zbiór, skopiować go lub usunąć, użytkownik wydaje systemowi operacyjnemu odpowiednie zlecenie, a ten przekłada je na język transmisji fizycznych, ścieżek i sektorów i wykonuje. Aby właściwie wypełniać swoje zadania, system gromadzi dane o wszystkich zbiorach i przechowuje je w wydzielonej części dysku, w tzw. kartotece (ang. directory). Wśród danych o każdym zbiorze znajdują się adresy wszystkich sektorów, zawierających treść zbioru, datę utworzenia, typ zawartości itp.

I znów mała dygresja. Zarówno sposób zapisu danych, jak i zawartość kartoteki, postać danych opisujących zbiory, rozmieszczenie tych danych na dysku bywają różne w różnych systemach operacyjnych. Dlatego zwykle niemożliwe jest bezpośrednie przenoszenie zbiorów danych z jednego systemu pod inny, nawet mimo korzystania z dyskietek tej samej wielkości. Stąd właśnie biorą się problemy przy próbie przeniesienia danych między różnymi maszynami. Np. dyskietka o średnicy 5.25 cala zapisana na Commodore C64 oczywiście da się włożyć do stacji 5.25 cala na IBM PC. Ale DOS na PC nie będzie w stanie dostrzec (ani przeczytać) żadnych danych, gdyż obie stacje używają różnych gęstości zapisu i różnie formatują dyskietkę.

Nawet wtedy gdy gęstości i fizyczne formaty są identyczne, to różnice w budowie kartotek występujące między różnymi systemami operacyjnymi uniemożliwiają odszukanie odpowiednich informacji i właściwą interpretację zawartości nośnika. Szczegółowe opisy budowy kartoteki i jej zawartości znaleźć można w dokumentacjach poszczególnych systemów.

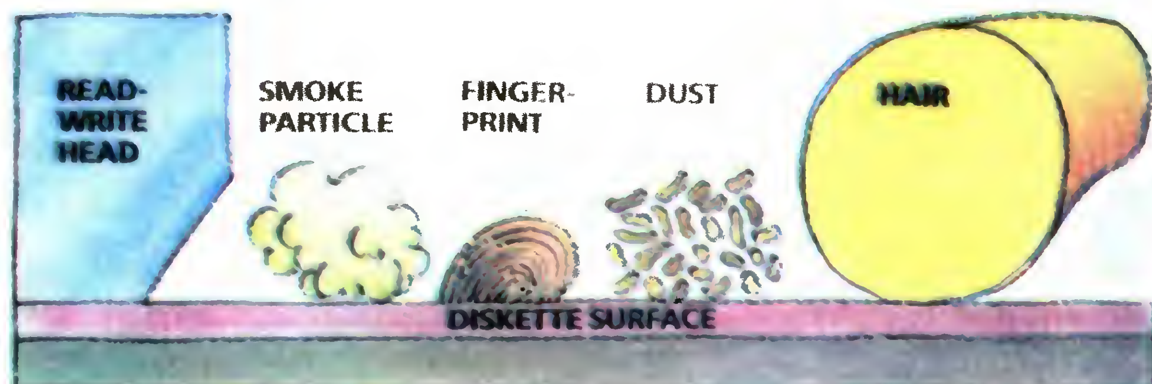
Niezależnie jednak od szczegółów realizacji podstawowe zasady są takie same w różnych systemach operacyjnych. Każdy zbiór musi być opisany w kartotece, każdy zbiór może zawierać wiele fizycznych sektorów. Użytkownika właściwie może nie obchodzić, gdzie znajdują się sektory, w których zapisano jego dane. Nie jest to jednak bez znaczenia dla szybkości działania maszyny.

Pamiętamy, że najbardziej czasochłonne są operacje przesuwania głowicy pomiędzy ścieżkami, więc dobry system będzie starał się przeznaczać na zbiór sektory leżące blisko siebie, tak aby wyszukiwanie kolejnych fragmentów zbioru (kolejnych sektorów) zajmowało możliwie najmniej czasu. Czy zawsze jest to możliwe, okaże się za chwilę, na razie zatrzymajmy się przy czytaniu kolejnych sektorów.

Aby to robić, musimy znać ich adresy, a więc przed rozpoczęciem operacji na zbiorze musimy odwołać się do kartoteki i odszukać tam jego opis. Operacja ta wymaga przesunięcia głowicy na ścieżkę zawierającą kartotekę, wczytania opisów zbiorów, odszukania właściwego i wreszcie odszukania początku zbioru, czyli ponowne przesunięcie głowicy, tym razem na ścieżkę zawierającą początek zbioru. Pobrany z kartoteki opis zachowujemy w pamięci operacyjnej, aby uniknąć dalszych czasochłonnych odwołań do kartoteki dysku. Skoro już mowa o tym, co trzeba zrobić przed rozpoczęciem czytania zbioru, to dodajmy jeszcze zarezerwowanie w pamięci operacyjnej miejsca na dane przesyłane z dysku na dysk, tzw. bufora (ang. buffer). Wszystkie te operacje noszą wspólną nazwę — otwarcie zbioru (ang. open). Operacja ta jest niezbędna, abyśmy mogli zrobić cokolwiek z zawartością zbioru, a więc zawsze musi poprzedzać czytanie czy pisanie. Raz otwarty zbiór pozostaje otwarty (a więc dostępny dla wszystkich operacji) aż do momentu operacji przeciwnej — zamknięcia zbioru (ang. close). Operacja ta umieszcza z powrotem w kartotece dane o zbiorze, mogą tam być nie tylko wartości wczytane z kartoteki przy otwarciu, ale również nowe wartości, np. adresy nowych, dołączonych do zbioru sektorów. Oczywiście, po zamknięciu zbiór znów jest niedostępny — aż do ponownego otwarcia.

Z powyższego opisu wynika, że otwarcie i zamknięcie zbioru jest operacją czasochłonną, między innymi dlatego, że wymaga kilkakrotnych przesunięć głowicy między ścieżkami. Pamiętajmy też, że w czasie gdy zbiór jest otwarty, nie mamy żadnej gwarancji, że zmiany w jego opisie znalazły odbicie w zapisanej na dysku kartotece. Uaktualniony opis zbioru jest zapisywany do kartoteki na dysk dopiero w momencie zamykania zbioru.

Spróbujmy wyciągnąć praktyczne wnioski z naszej nowej wiedzy. Skoro każdy otwarty zbiór zajmuje dla siebie pewien obszar pamięci operacyjnej, to otwarcie wielu zbiorów może zająć jej zbyt wiele i zdezorganizować pracę maszyny. Dlatego większość systemów operacyjnych ogranicza liczbę zbiorów, które równocześnie mogą być otwarte. Niektóre systemy, np. MS DOS na IBM PC, pozwalają użytkownikowi zmienić przyjęte przez system ograniczenie. Z możliwości tej trzeba zwykle skorzy-



stać, gdy program pada z sygnalizacją: „Too many files are open".

Skoro informacje o zmianach wprowadzonych do zbioru są wpisywane do kartoteki dopiero w momencie zamykania zbioru, to bardzo niebezpieczne jest przerywanie programu piszącego na dysk przed jego normalnym zakończeniem (czasami takie przerwanie się wykonuje się wbrew naszym intencjom — przez zanik zasilania w sieci elektrycznej). Efektem takiego przerwania może być sytuacja, gdy zawartość zbioru na dysku została drastycznie zmieniona, np. dopisano nowe dane zajmujące nowe sektory, ale opis zbioru w kartotece nie został odpowiednio zmodyfikowany i nie odpowiada nowej zawartości. Grozi nam wtedy strata świeżo dopisanych danych, w skrajnym wypadku może nastąpić całkowita dezorganizacja i oczywiście strata **całego** zbioru lub **nawet** innych zbiorów.

I wreszcie coś dla początkujących programistów. Skoro otwarcie i zamknięcie zbioru jest tak czasochłonne, nie należy wykonywać go bez potrzeby, w szczególności, jeśli to tylko możliwe, nie należy umieszczać tych operacji wewnątrz pętli! Zilustrujmy to bardzo prostym przykładem. Jeśli w programie chcemy wykonać sto razy odczyt ze zbioru, to rozwiązanie:

```
FOR I=1 TO 100
    OPEN zbiór
    READ
    CLOSE zbiór
NEXT I
```

jest fatalne i będzie się wykonywać wielokrotnie dłużej niż:

```
OPEN zbiór
FOR I=1 TO 100
    READ
NEXT I
CLOSE zbiór
```

Powyższy przykład nie jest jedynym możliwym. Jest o wiele więcej sytuacji, w których niewłaściwe programowanie zmusza pamięć do dużo większej pracy niż wynosi niezbędne minimum.

Sygnalizowany tutaj problem ma znaczenie nie tylko dla programistów. Warto o nim pamiętać korzystając z gotowych programów użytkowych. Np. zapoznając się z bazą danych, której mamy zamiar używać, warto zastanowić się, które z jej funkcji będą wymagały otwarcia lub zamknięcia zbioru. Uzbrojeni w taką wiedzę, możemy zoptymalizować odwołania do danych i zaoszczędzić sporo czasu.

Nawet optymalnie zorganizowane odwołania do pamięci dyskowej są realizowane dużo wolniej niż wykonywane przez procesor odwołania do pamięci operacyjnej czy operacje arytmetyczne. Konkretne liczby zależą od wyboru porównywanych urządzeń, ale generalnie można powiedzieć, że pamięci dyskowe pracują tysiące razy wolniej niż półprzewodnikowe pamięci operacyjne mikrokomputerów. Jeśli więc chcemy, aby nasze programy działały możliwie najszybciej, musimy ograniczać kontakty z pamięcią zewnętrzną do niezbędnego minimum. I znowu przykład, z braku miejsca dość prymitywny. Załóżmy, że te same dane z dysku są potrzebne w trzech różnych miejscach programu. Na pewno musimy przeczytać je za pierwszym razem. Jeśli jednak znajdziemy miejsce w pamięci operacyjnej i tu przechowamy przeczytane dane, to w ten sposób unikniemy dwóch następnych odczytów z dysku.

A teraz pytanie: czy szybkość wykonywania operacji przez pamięć dyskową zależy tylko od użytkownika i czy jest ona zawsze taka sama? Słusznie podejrzewacie, że skoro stawiamy takie pytanie, to odpowiedź może być interesująca. Otóż może się zdarzyć, że mając ten sam komputer, ten sam program i to samo urządzenie pamięci będziemy otrzymywać różne czasy zapisu takiego samego zbioru. Jeśli weźmiemy np. duży twardy dysk, to w miarę upływu czasu średnia prędkość wykonywania zapisu i odczytu zbiorów może zmaleć. Nie jest to jednak spowodowane starzeniem się materiałów czy zmniejszeniem prędkości obrotowej silnika.

Jak pamiętamy, bardzo czasochłonną fazą pracy pamięci jest przesuwanie głowicy między ścieżkami, straty czasu powoduje też czekanie na właściwy sektor na ścieżce. Dlatego przydzielając miejsce na nowy zbiór, system operacyjny stara się przeznaczyć na to sektory tak położone, aby po przeczytaniu jednego rozpoczęcie czytania następnego nie wymagało żadnych czasochłonnych ruchów głowicy. Upraszczając nieco szczegóły działania powiemy, że system stara się przydzielać na zbiór sektory sąsiadujące ze sobą. Gdy dysk jest nowy i pusty, udaje się to z łatwością — każdy nowy zbiór zapisujemy bezpośrednio za poprzednim i całe wolne miejsce na dysku (pula wolnych sektorów) stanowi jeden spójny obszar. Kiedy jednak dysk zapełnia się, obszar ten zostaje rozdzielony, a jedyną możliwość zapisania nowych zbiorów stanowią sektory odzyskane po zbiorach usuniętych przez użytkownika. Te wolne fragmenty są jednak rozrzucone po całym dysku. Idealnie jest, jeśli chcąc zapisać zbiór np. o długości 10KB, znajdziemy miejsce po zbiorze, który miał właśnie dokładnie 10KB. Jeśli jednak (co bardzo prawdopodobne) takiego miejsca nie ma, to albo musimy przeznaczyć na nasz zbiór kilka mniejszych, odległych od siebie fragmentów albo jeden większy od 10, np. 12KB. W tym drugim wypadku zostaje nie wykorzystany fragment 2KB. W miarę upływu czasu sytuacja na intensywnie eksploatowanym dysku znacznie się pogarsza, gdyż wolne miejsce odzyskiwane po usuwanych zbiorach dzieli się na coraz mniejsze kawałki (tak jak w powyższym przykładzie: z fragmentu 12KB został fragment 2KB). Takie zjawisko nazywane jest fragmentacją wolnego miejsca na dysku. Jego następstwem jest konieczność zapisywania nowych zbiorów nie w jednym miejscu, lecz w wielu, często odległych od siebie fragmentach. Oczywiście zapis i odczyt tak zorganizowanego zbioru zajmuje znacznie więcej czasu niż przy zapisie w jednym spójnym obszarze. W ten sposób tajemnica zmiany prędkości tego samego dysku uzyskuje wyjaśnienie nie burzące praw fizyki.

Zwróćmy też uwagę, że pojawienie się fragmentacji nie jest spowodowane żadnym błędem w sztuce czy czyjąś złą wolą. Jest ona obiektywnym następstwem przyjętego systemu organizacji zbiorów na dysku i dynamicznego dopisywania i usuwania danych.

Skoro fragmentacji nie można uniknąć, to czy przynajmniej można zlikwidować jej następstwa? Oczywiście tak. Trzeba przepisać zbiory na dysku, tak aby następowały bezpośrednio po sobie, a wszystkie wolne sektory zajmowały jeden obszar. Oczywiście najłatwiej to zrobić, jeśli dysponujemy drugą pustą pamięcią, na którą możemy przepisać wszystkie zbiory, jednak są również programy reorganizujące zawartość pamięci w ramach jednego dysku.

Tak się złożyło, że fragmentacja jest ostatnim opisanym w tym artykule problemem dyskowym. Nie chciałbym, żeby ktoś odniósł przez to wrażenie, iż jest ona dominującym problemem dla użytkownika mikrokomputera wyposażonego w pamięć na dyskietkach lub twardym dysku. W tym kręgu użytkowników fragmentacja pozostaje raczej ciekawostką, gdyż codzienne obserwacje wskazują, iż dużo więcej kłopotów sprawia przypadkowe skasowanie potrzebnych zbiorów lub uszkodzenie zapisu i brak zapasowej kopii. Dlatego, choć rozważania o dyskach są ważne i potrzebne, warto pozostać na ziemi choć jedną nogą i postarać się nie polewać dyskietek herbatą.

Andrzej Krul

KIESZONKOWY —IBM PC—

Reklama lubi przesadę, ale tym razem, wbrew pozorom, w naszym tytule nie ma ani reklamy, ani przesady. Trzeba tylko uzupełnić, że producentem maszyny zgodnej z IBM PC, ale będącej wielkości kasety wideo VHS, jest niedawno powstała firma Poqet Computer Corporation.

Skoro sedno sprawy odkryliśmy już w tytule, więc jedyne co nam pozostaje to podać szczegóły techniczne tej rewelacji. Najpierw, aby uniknąć niejednoznaczności, dokładne wymiary zamkniętej maszyny: 23.5x222.3x1 x109.2mm. Wyświetlacz ciekłokrystaliczny, o wymiarach 173x69mm (przekątna ekranu ok. 7 cali), pracuje w trybie znakowym 80 kolumn na 25 wierszy lub graficznym (tryb CGA lub MDA) z rozdzielczością 640 na 200 punktów. Jak większość laptopów, Poqet PC ma otwieraną do góry pokrywę, na jej wewnętrznej stronie znajduje się widoczny po otwarciu ekran.

Klawiatura zawiera 77 klawiszy, część z nich pełni podwójną funkcję, pozwalając uzyskać klawiaturę numeryczną. Klawisze są spore — o powierzchni ok. 3 cm^2.

Wnętrze mieści mikroprocesor Intel 80C88, taktowany zegarem 7MHz (przypomnijmy, podstawowy IBM PC miał zegar 4.77MHz, wersja turbo 8MHz). Układy zawierające BIOS oraz pozostałe współpracujące z procesorem kości zostały specjalnie zaprojektowane przez firmę Poqet. Pamięć maszyny — to 512 kilobajtów RAM oraz, uwaga, 640 kilobajtów ROM. W pamięci ROM zapisany jest system operacyjny MS-DOS 3.3, GWBASIC oraz zestaw podstawowego oprogramowania użytkowego opracowanego przez firmę Poqet. W zestawie tym znalazły się: kalkulator, prosty edytor, kalendarz do rejestracji terminów (przypominający o ich upływie), program komunikacyjny przygotowany do współpracy z modemem (modem nie wchodzi w skład komputera), notatnik adresowy, program do zarządzania zbiorami i wreszcie program pozwalający przesyłać zbiory. Do wyboru jednej z powyższych funkcji służy wyświetlane na ekranie menu. Dzięki tym programom bardzo wiele można zrobić, mimo braku w maszynie stacji pamięci dyskowej. Pamięć operacyjna maszyny jest zasilana przez cały czas, dzięki czemu pozostawione w niej dane nie giną. Dodatkową zaletą zapisania programów aplikacyjnych w pamięci ROM jest duża szybkość ich uruchamiania — większa niż w przypadku wykorzystania pamięci dyskowych. I wreszcie, skoro wykonanie programów nie wymaga uprzedniego załadowania ich do pamięci operacyjnej, to większość pamięci RAM można przeznaczyć na dane.

Do przechowywania i przenoszenia danych oraz programów zastosowano karty pamięci. Urządzenie pozwala korzystać równocześnie z dwóch kart. W zależności od typu karta może mieścić od 32K do 512K bajtów. Dostępne będą dwie wersje kart: ROM, z zapisanymi programami użytkowymi, oraz RAM, pozwalające zapisywać dane. Istotne jest, że również programy dostarczane na kartach są wykonywane bez przepisywania (ładowania) do pamięci operacyjnej, a więc nie ograniczają miejsca na dane. Karty te mają jednak przynajmniej dwie wady. Pierwsza z nich to cena — ok. 1 dolara za kilobajt, czyli ok. 500$ za kartę 512K. Jest jednak szansa, że przy dużym zainteresowaniu i masowych dostawach cena ta spadnie. Już teraz producenci oprogramowania, którzy zechcą dostarczać swe produkty na kartach, mogą hurtowo kupować karty po 50 dolarów sztuka. Tu jednak pojawia się drugi problem. Programy muszą zostać zmodyfikowane tak, aby mogły działać bezpośrednio z pamięci na karcie. Niemniej jednak główny programista firmy Poqet Ian Cullimore twierdzi, iż zainteresowanie firm softwareowych przeniesieniem programów jest bardzo duże. Swoje zaangażowanie potwierdził m.in. Lotus. Dla producentów oprogramowania dużą zaletą jest wykonywanie programów bezpośrednio z karty, co ogranicza piractwo.

Firma zapowiada również na najbliższe miesiące (w chwili, gdy to czytacie, już pewnie jest dostępne) urządzenie podłączane do zwykłego PC, pozwalające przepisywać dane na karty pamięci.

Natomiast już teraz w skład standardowego wyposażenia wchodzi złącze i kabel pozwalające łączyć Poqet PC bezpośrednio z szyną IBM PC lub z urządzeniami peryferyjnymi. Transmisja danych jest szeregowa (RS232C), maksymalna prędkość 19200 bitów na sekundę.

I wreszcie wielka niespodzianka — zasilanie urządzenia. Mieszczące się w maszynie baterie zapewniają 100 (słownie sto) godzin pracy. Jest to możliwe głównie dzięki redukcji (dziesięciokrotnej w stosunku do powszechnie stosowanych dziś rozwiązań) zużycia energii przez maszynę, przede wszystkim zaś przez wyświetlacz. Ponieważ rozwiązanie jest zgłoszone do opatentowania, żadnych bliższych szczegółów na razie nie podano. Zasilacz jest wyposażony w kondensator podtrzymujący zasilanie przez minimum 5 minut, umożliwiający bezbolesną wymianę baterii. Karty pamięci wyposażone są we własne baterie, podtrzymujące zawartość pamięci po odcięciu od zasilania przez okres od roku do dwóch lat.

Omówiony zestaw, tzn. maszyna mająca 512KB RAM + 640KB ROM, wbudowane system operacyjny i oprogramowanie oraz złącze z kablem, kosztuje 2000 dolarów. Dodatkowo można dokupić bateryjną stację dyskietek 3.5 cala, o pojemności 1.44MB. Bateria wystarcza na 20 godzin pracy. Cena 395 dolarów.

Prezentowany komputer prawie na pewno stanie się dużym przebojem. Niezależnie jednak od tego, co pokaże przyszłość, zastosowane w nim rozwiązania techniczne są naprawdę rewelacyjne.

Na podstawie „Byte" opracował (pan)

Zaczynamy!

W piątym roku istnienia Bajtka do czterech klanów, stanowiących podstawę każdego numeru, dołączamy piąty — klan IBM-a.

Czy taki klan jest potrzebny? Odpowiedzi na to pytanie udziela kilkaset tysięcy komputerów XT i AT, używanych we wszelkich instytucjach (i coraz częściej w domach). Kiedy na rynku dominowały komputery ośmiobitowe, pisaliśmy głównie o nich, teraz nadszedł czas szesnastobitowców i dla nich też trzeba znaleźć miejsce. Czy trzeba go szukać w „Bajtku"? Z mikrokomputerami mam do czynienia od sześciu lat. W 1984 roku korzystałem w pracowni uniwersyteckiej z Neptuna — ośmiobitowca z interpreterem Basic-a. W rok później dorobiłem się ZX 81, a po kilku miesiącach Spectrum. W 1988 roku siadłem do pracy przy IBM PC — najpierw XT, potem AT, a od stycznia korzystam z Super AT — komputera opartego na procesorze 80386. Nie piszę tego, żeby się chwalić — podejrzewam, że nie ja jeden przeszedłem taką drogę, a jestem pewien, że wielu naszych czytelników właśnie po niej idzie. Dla nich, żeby nie zabłądzili, będziemy pisać.

Klan IBM-a miał już kiedyś swoją inaugurację, w sierpniowym numerze z 1987 roku. Był to jednak zupełny falstart. W tekście znalazło się między innymi zdanie: „... praca na komputerach klasy IBM PC ... polega na pracy osobistej, doraźnej, raczej chwilowej niż długotrwałej". Taki też był charakter tamtego klanu, który urwał się po dwóch numerach. Mam nadzieję, że tym razem będzie lepiej, zwłaszcza że w Redakcji jest kilka osób, które pracują przy komputerach PC raczej długotrwale, po kilkanaście godzin dziennie, niż chwilowo. Doszło do nieco paradoksalnej sytuacji, w której komputer AT służy do pisania o Spectrum. Żeby przybliżyć stan równowagi, zaczniemy pisać o AT.

Co chcemy robić w nowym klanie? To samo co we wszystkich pozostałych — proponować, tłumaczyć, pomagać i prezentować. Nie musimy startować od zera (jest już trochę całkiem niezłych książek), ale też nie możemy zakładać, że wszyscy wszystko wiedzą. Postaramy się pomóc i tym, którzy dopiero usiedli przed klawiaturą, i tym, którzy już się z nią zżyli. Mamy w planach trudne têksty, o poważnych problemach, i łatwe, pozwalające na chwilę relaksu. Liczymy na pomoc czytelników — zarówno w postaci uwag na temat naszego klanu, jak i materiałów do druku.

Z pewnych powodów redagowanie klanu IBM-a będzie trudniejsze od pozostałych. Podstawowy to taki, że IBM PC jest sprzętem profesjonalnym, z bardzo bogatą biblioteką poważnych programów użytkowych, i rzadko służy do zabawy w programowanie. Opisywanie pracy z jakimkolwiek programem oznacza ułatwienie korzystania z niego potencjalnym złodziejom, a jest to ostatnia rzecz, jaką mamy zamiar robić. W przypadku innych klanów sprawa wygląda prościej — program w Basic-u na Spectrum może być świetnym materiałem do druku, nie wywołując niczyjego oburzenia. Miejmy nadzieję, że uda nam się znaleźć odpowiednią formułę, satysfakcjonującą wszystkich — i użytkowników, i twórców oprogramowania (nie tylko krajowych!).

Zaczynamy skromnie, ale ze smakiem. Przedstawiamy dwa programy: edukacyjny PC Globo i edytor MSWord 5.0. Nasz redakcyjny fachowiec wypowie się na temat gier — bo i do tego może służyć IBM.

Za miesiąc — bierzemy się za programowanie.

Marcin Borkowski

MICROSOFT WORD — czy tylko dobry edytor?

Word jako edytor tekstu nie zdobył na polskim rynku takiej pozycji jak choćby ChiWriter, chociaż pod wieloma względami jest lepszy. Istnieją jednak co najmniej trzy jego przeróbki, z polskimi literami na ekranie, drukarce i klawiaturze, co świadczy o tym, że został przez naszych rodzimych „programistów" dostrzeżony i doceniony.

Najnowsza wersja jest obdarzona numerem 5.0. Zanim jednak napiszę, co mi się w niej najbardziej podoba, krótka prezentacja wcześniejszej wersji — 4.0. Krótka — bo dokładniejszy opis tego, co może Word, zająłby cały numer „Bajtka".

Prezentację zacznijmy od wbudowanych sposobów formatowania tekstu. Można formatować znaki, które pojawią się na wydruku: wybierać krój czcionek, wytłuszczenie, podkreślenie, kursywę itp. Liczba krojów, jaką mamy do dyspozycji, zależy od używanej drukarki, gdyż Word nie dysponuje (jak ChiWriter) własnymi wzorcami czcionek. Można za to w opisie drukarki zdefiniować ciągi kodów wysyłanych w określonych sytuacjach do drukarki (np. przy włączaniu trybu NLQ, albo zamiast konkretnego znaku ASCII). Dzięki temu każda litera może zostać zastąpiona dowolnym znaczkiem. Istnieje również możliwość korzystania z krojów ładowanych do drukarki przed ich użyciem (download fonts).

Wróćmy jednak do formatowania. Można odpowiednio przygotować wygląd strony, definiując jej rozmiary i miejsce drukowania numeru (o ile zdecydujemy się na numerowanie stron), a także liczba szpalt (niekoniecznie na całej stronie — może to być tylko jej fragment). Można wreszcie formatować wybrane akapity, ustalając lewy i prawy margines, sposób justowania, wcięcie pierwszej linii, tabulator i kilka innych parametrów. Wszystkie powtarzalne wzory akapitów, stron i znaków można zapamiętać w galerii wzorców, tak że użycie ich wymaga naciśnięcia tylko dwóch lub trzech klawiszy.

Przy opracowywaniu tekstu można posługiwać się myszą. Wydaje się to śmieszne — po co mysz do pisania, ale w trakcie edycji można przy jej pomocy wykonać bardzo wiele operacji. Po kilku minutach pracy okazuje się, że wybieranie przy jej pomocy fragmentów tekstu i poruszanie się po dokumencie jest znacznie szybsze i wygodniejsze.

Word potrafi posortować listę zawierającą liczby lub nazwy (np. listę nazwisk), potrafi ją również ponumerować. Można przy jego pomocy włączyć do tekstu dane zawarte w arkuszu obliczeniowym, przygotowanym przez inny program (Lotus 1-2-3, Multiplan, Excel). Do dyspozycji piszącego są słowniki wyrazów bliskoznacznych i ortograficzny. Przed wydrukowaniem tekstu można przygotować indeksy, spisy treści i tabel lub ilustracji, a także poprzenosić długie wyrazy — Word proponuje, które słowa przenieść i w którym miejscu. Niestety — z naszego, polskiego punktu widzenia — Word „zna" tylko angielski.

Do każdego dokumentu można przypisać krótką informację, w której będzie zapisane kto i kiedy dokument przygotował, data ostatniej poprawki, a także wybrane słowa kluczowe. Następnie na podstawie tych informacji można w każdej chwili wybrać spośród znajdujących się na dysku zbiorów te, które z jakichś powodów są nam akurat potrzebne. Jeżeli chcemy porównać aktualnie pisany tekst z innym, możemy pracować z obydwoma (trzema, czterema...) naraz, każdy z nich będzie w innym okienku.

Kolejną, bardzo wygodną możliwością edytora jest pisanie na podstawie przygotowanego wcześniej szkicu (coś w rodzaju spisu treści). Co ważne, kiedy napiszemy już część tekstu i uznamy, że szkic był nie najlepszy, możemy go poprawić, a wynikające z poprawek zmiany zostaną automatycznie uwzględnione w dokumencie.

Inne, bardzo wygodne opcje — to rysowanie tabelek przy pomocy wybranej wcześniej linii (pojedyncza, podwójna, mieszana); makra, czyli zapamiętane, często używane sekwencje klawiszy (dzięki nim można się uporać z problemem polskich liter), pozwalające, dzięki kilku dodatkowym instrukcjom, na pisanie programików przeformatowujących tekst; automatyczne numerowanie przypisów; mail-merge, czyli drukowanie serii analogicznych dokumentów, nieznacznie różniących się między sobą (np. nazwiskiem i imieniem adresata); drukowanie w tle (podczas pracy z następnym dokumentem); nagłówki i stopki (taki sam tekst na górze lub u dołu każdej strony); wypełnianie formularzy i wiele innych.

Ktoś może powiedzieć — phi, inne edytory też to potrafią — i będzie miał rację. Kilka lat temu, kiedy edytorów było niewiele, można było je porównywać pisząc, że jeden może to, a drugi tamto. Dzisiaj takie porównania straciły sens. Poza pewnymi rzeczywiście specjalnymi opcjami, figurującymi tylko w nielicznych, specjalizowanych edytorach — jak składanie wzorów matematycznych lub chemicznych w ChiWriterze — większość sposobów usprawnienia pracy piszącego jest dziś standardem, bez którego edytor przegrywa w konkurencji z innymi. Microsoft ze swoim produktem na razie nie musi obawiać się wypadnięcia z rynku.

Teraz już konkretnie o prawie najmłodszym dziecku Microsoft-u, czyli o wersji Word 5.0 (najnowsza wersja nosi nazwę Word for Windows i jest w pewnym sensie zupełnie nowym programem). Od wszystkich wcześniejszych różni się on jedną podstawową cechą — nie pracuje na komputerach klasy XT, wymaga bowiem procesora 80286 lub 80386. Ma to ścisły związek z pewną nietypową jak na razie cechą — edytora można używać nie tylko pod kontrolą MS DOS-u, ale również systemu OS/2.

Nie próbując dokładnie omówić nowych możliwości, zatrzymam się na trzech, które okazały się być dla mnie najbardziej przydatne. Przede wszystkim **Help**. Nareszcie sprawę potraktowano poważnie. Do tej pory pomoc, jakiej Word udzielał, była skąpa i nie pozwalała na wygodne korzystanie z niego bez sięgania do instrukcji. Teraz sytuacja się zmieniła. Wszystkie opcje są precyzyjnie opisane i chociaż od czasu do czasu użytkownik jest odsy-



```
1═══0·········1·········2·········3·········4·········5·········6·········7·········8···
   ostateczności.

SP     W ramach ułatwiania życia użytkownikom wzięto się  także
   za tych, którzy w  swoich  tekstach  potrzebują  ilustracji.
   Wprowadzono  opcję  Link Graphics,  czyli  włączanie  do
   dokumentu grafiki. Rysunek,  przygotowany  wcześniej,  można
   wprowadzić w niemal dowolne miejsce strony.  Word  akceptuje
   kilka  formatów  rysunków,  przygotowywanych   przy   pomocy
   różnych programów - np.  PaintBrush,  Lotus,  ilustracje  ze
═══════════════════════════════════════════════════════════════════════KLAN1.DOC═
2═══[···]·····1·········2·········3·········4·········5·········]·········7·········8···

SP     Teraz trzeba  uzbroić  się  w  cierpliwość  i  uruchomić
   prezentowany program. Na Spectrum sto cząstek ulega redukcji
   w ciągu jednej godziny,  toteż  żeby  móc  obejrzeć  ciekawe
   efekty swojej pracy trzeba je włączyć  conajmniej  na  dobę.
   Niecierpliwym radzę obejrzeć od razu  rysunki  które  w  tym
   celu przygotowałem. Widać na nich,  że  im  łatwiej  cząstki
   wpadające  na  elektrodę  ulegają  redukcji,  tym  bardziej
═════════════════════════════════════════════════════════════════════W_CHEMII.DOC═
COMMAND: Copy Delete Format Gallery Help Insert Jump Library
         Options Print Quit Replace Search Transfer Undo Window
Microsoft Word Version 5.0
P3 D1 L34.6 C61   {}                                        Microsoft Word
```

łany do książki, jako najobszerniejszego źródła wiedzy, to dzieje się tak tylko w ostateczności.

W ramach ułatwiania życia użytkownikom wzięto się za tych, którzy w swoich tekstach potrzebują ilustracji. Wprowadzono opcję **Link Graphics**, czyli włączanie do dokumentu grafiki. Rysunek, przygotowany wcześniej, można wprowadzić w niemal dowolne miejsce strony. Word akceptuje kilka formatów rysunków przygotowywanych przez różne programy (PaintBrush, Lotus 1-2-3, ACAD) lub skanera. Można również włączyć w tekst rysunek przygotowany w postaci zbioru przeznaczonego do wysłania bajt po bajcie na drukarkę — trzeba tylko podać rozmiary ilustracji, tak by możliwe było właściwe złamanie strony. Dodatkowo, dla korzystających z innych niż wymienione programów, razem z edytorem sprzedawany jest rezydentny program o nazwie **capture**. Służy on do nagrywania na dysk (w odpowiednim formacie, rozpoznawanym przez edytor) grafiki znajdującej się na ekranie w momencie naciśnięcia klawisza PrtScr.

Drugą bardzo wygodną opcją jest **preViev**, czyli podglądanie wydruku zanim zostanie on zrobiony. Znakomicie zmniejsza to ilość papieru straconego na próby ustawienia tekstu we właściwym miejscu strony, tak by nie zostało za dużo wolnego u góry ani u dołu itd. Można również precyzyjnie wkomponować rysunek w tekst, nie mordując nadmiernie drukarki.

Pozostałe ulepszenia polegały na wzbogaceniu istniejących już wcześniej możliwości i usunięciu błędów, utrudniających pracę z wersją 4.0 (z moich doświadczeń wynika, że można pracować przez kilka miesięcy z edytorem nie wiedząc o ich istnieniu, wszystko zależy od opcji, z których się korzysta).

Przedstawione możliwości prowokują pytanie — czy najnowsza wersja Worda to jeszcze rozbudowany edytor, czy już skromny program typu Desk Top Publishing? Skłaniam się ku temu drugiemu, choć zdaję sobie sprawę z dyskusyjności tego poglądu. Stworzenie gazety składającej się z różnych tekstów rozmieszczonych na jednej kolumnie, jest bez korzystania z różnych niewygodnych sztuczek niemożliwe, i to świadczy przeciwko traktowaniu edytora jako DTP. Z drugiej strony, napisanie i wydrukowanie porządnie wyglądającej instrukcji czy książki jest proste i wygodne, zwłaszcza gdy ma się do dyspozycji dobrą drukarkę. Wszystko zależy więc od tego, co chcemy publikować — może się okazać, że Word 5.0 jest dla nas zupełnie wystarczający.

Mimo bardzo wysokiej oceny, jaka należy się opisywanemu edytorowi, nie jest to program bez skazy. Nie udało się w trakcie jego pisania opanować do końca kłopotów z wyświetlaniem obrazu. Choć nie powoduje to strat w pisanym tekście (o ile wyjdziemy z programu „w ciemno", nagrywając po drodze redagowany tekst), nie ułatwia to także pracy. Dzieje się tak zwłaszcz... po awarii drukarki. Cóż, prawo Murphy'ego mówi, że każdy uruchomiony i działający program zawiera przynajmniej jeden błąd...

Marcin Borkowski

```
PRINT PREVIEW: Exit Jump Options Print
Use PgUp and PgDn to scroll through document
2 D1    P3 D1                                          WORDGRAF.DOC
```

GRY



DLA WAPNIAKA

Nikt nie powie, że IBM służy do gier. Atari, Spectrum — tak, ale IBM? Każdy widział Diggera lub Tetris, ale kto zdawał sobie sprawę, że to jedynie okruch rozrywkowego oprogramowania IBM-a.

Kto pisze te gry i dla kogo? Piszą na pewno firmy softwarowe, ale dla kogo, to trudno nam sobie wyobrazić. W Polsce IBM jest i długo pozostanie symbolem luksusu i postępu technicznego, dostępnym wybrańcom losu oraz businessmanom. Natomiast w krajach zwanych zachodnimi nawet AT znajduje się w każdym domu i nieczęsto obarczane jest ambitnymi obliczeniami. Naturalny więc pęd zastępców młodocianych i podstarzałych amatorów rozrywki i sportu w fotelu tworzy ogromny popyt na gry.

Gdy na rynku np. angielskim pojawia się nowa gra, to z reguły od razu w kilku wersjach: na Spectrum, Commodore, Amstrad, Amigę, Atari ST oraz IBM PC. Wersje te różnią się między sobą w różnym stopniu, zależy to od wysiłku włożonego w ich przygotowanie. Najczęściej jednak jest tak, że wersja na IBM jest najpełniejsza. Na nim powstaje bowiem pierwowzór i... pozostaje, gdy w okrojonej i nieco zubożonej postaci instalowany jest na mniejszych komputerach. One wykazać się mogą jedynie lepszą oprawą muzyczną, najczęściej robioną oddzielnie.

Nie jest to jednak regułą. Istnieje bowiem sporo gier przeniesionych na IBM np. z Commodore. Aby nie bawić się w szczegóły, zaemulowano Commodore na IBM i uruchomiono nań „przepchniętą" złączem szeregowym grę. Z ostatnich ciekawostek wymienić można emulator ZX Spectrum, Commodore, Amstrada, Amigę, Atari ST oraz możliwości. Programowanie, uruchamianie gier i programów użytkowych, współpraca z urządzeniami zewnętrznymi jest taka, jak na Spectrum.

Jasne jest, że zaadaptowana na IBM gra „spod" jest uboższa niż oryginał. Grafika jest taka jak pierwowzoru, często pozbawiona części kolorów. Takie przykłady to Dam Busters, Ninja, Destroyer, Pitstop, Karateka.

Jeśli zaś gra powstała na IBM-ie, to w swym rodzinnym otoczeniu wygląda tak, jak życzyli sobie tego producenci. Czyli z reguły doskonale.

Z uwagi na rozpowszechnienie różnych kart grafiki z komputerami IBM, coraz więcej gier posiada opcje wyboru karty. Hercules daje grafikę monochromatyczną, za to bardzo wysokiej rozdzielczości. CGA to cztery kolory w marnej rozdzielczości. EGA cieszy oczy 16 kolorami i wysoką rozdzielczością, a hit ostatniego roku — VGA — bije ST, Amigę i co tam kto jeszcze chce.

Podstawowe atrybuty IBM-a to szybkość i „ładowność". Dlatego też nie ma kłopotów z przetwarzaniem obrazu ani animacją, niektóre gry chodzą wręcz za szybko. Dla nich przyda się programowy spowalniacz pracy komputera, obrzydzi on jednak ewentualną współpracę z dyskiem.

Problemy szybkościowe jednak istnieją. Dla przykładu, XT z kartą EGA pracuje już ślimaczo, gdyż potrzebuje ona 256K RAM na bieżąco i przetwarzanie jest bardzo złożone. Podobnie wolniejsze modele AT z VGA — momentami nie dają sobie rady.

Jako że gros IBM-ów znajduje się wciąż w miejscach pracy, zaś oddawanie się graniu jest niebezpieczne z uwagi na możliwość „łapanki", wielu producentów wyposaża gry w tzw. Boss Key („sposób na szefa"). Jest to klawisz lub kombinacja klawiszy, które powodują wyświetlenie objawów ciężkiej pracy — skomplikowanego wykresu, „lewej" płachty z Lotusa lub po prostu napisu „proszę czekać, obliczenia".

Niektóre firmy robią sobie bardziej długofalowe żarty, np. w momencie powrotu z Boss Key pojawia się komunikat „O rany, wszystko zapomniałem" i grać trzeba od początku.

Gry na IBM-ie to sama przyjemność. Amatorzy złożonych tekstówek mogą wyżywać się na nieskończonej serii „Animated Adventure" firmy Sierra On-Line: Larry, Space Quest, Police Quest, King's Quest. Piloci próbować swych w GunShip-ie, F-15 i F-18, Falconie. Urodzeni kapitanowie grają w Silent Service, 688 Attack Sub, Hunt for Red October, Strike Fleet. Ambitni męczą Hostages i Defender of the Crown. Stratedzy bawią się City Simulation, Populous, Ogre. Dla miłośników gier konwencjonalnych pozostaje brydż, szachy w wielu odmianach, bilard i wiele innych.

Na koniec pozostawiłem nieśmiertelną kwestię zdobywania gier. Co sprytniejsi zbierają, co im w ręce wpadnie, i sprzedają swój dorobek potrzebującym. Dużo powszechniejszym sposobem jest jednak rozdawanie nowości znajomym. Korzystają na tym wszyscy. A najbardziej firmy piszące programy antywirusowe, gdyż gry są najczęstszymi nosicielami komputerowej zarazy.

Na razie nie ma innego sposobu rozchodzenia się programów niż ich naturalna dyfuzja w skomputeryzowanym społeczeństwie. Nikt nie oczekuje, by właściciele komputerów żyli cnotliwie, nie kalając się pirackimi kopiami. Ale „Kalemu ukraść — źle", więc zwracajmy uwagę na polskie produkty.

Nie można dopuścić do sytuacji, byśmy byli sobie wrogami. Możliwe, że gdy uchwalona zostanie wyczekiwana ustawa, statystycznego Polaka stać będzie na jedną marniutką strzelaninkę na kwartał. Jeszcze gorzej będzie z programami użytkowymi. Na razie jednak grajmy, bo naprawdę warto!

Marcin Przasnyski

PC GLOBE

KOMPUTEROWY ATLAS ŚWIATA

Brak regulacji prawnych w Polsce w dziedzinie własności programów jest przyczyną ubóstwa rynku oryginalnego polskiego oprogramowania.

Z tego powodu większość opisywanych w naszych czasopismach programów to produkty firm zachodnich. Jednym z nich jest właśnie program PC Globe, prezentowany na targach Cebit 90 przez amerykańską firmę PC Globe Incorporation. Do Redakcji dotarła tylko jego dość okrojona wersja demonstracyjna, ale mimo to uważamy, że dzięki olbrzymim walorom edukacyjno-informacyjnym jest ona godna dalszego opisu.

Program PC Globe jest skrzyżowaniem Atlasu Świata z dużą bazą danych, zawierającą szczegółowe informacje ekonomiczne, polityczne i geograficzne o wszystkich krajach kuli ziemskiej.

Menu główne

Menu główne programu, przedstawione na rysunku 1, zawiera 7 pozycji. Pierwsza z nich — *Help* — pozwala uzyskać wszelkie niezbędne informacje na temat obsługi programu. Trzy kolejne opcje — *World, Region* i *Country* — umożliwiają wybór obiektu (kontynent, państwo lub miasto) na poziomie mapy świata, wybranego kontynentu lub państwa. Dodatkowe informacje dotyczące danego kraju dostępne są za pomocą pozycji *Database*. Opcja *Utilities* grupuje pewne specjalne operacje ogólnej użyteczności. Opuszczenie programu odbywa się poprzez wybór ostatniej pozycji menu głównego.

Korzystanie z programu

Sposób wykorzystania programu przedstawimy na przykładzie kilku ekranów przeniesionych na drukarkę za pomocą klawisza PrtScr i zainstalowanego wcześniej pliku systemowego GRAPHICS.COM. Efekt wyboru z pozycji menu *Region*, kontynentu Ameryka Pd., państwa Brazylia i miasta Rio de Janeiro widać na rys. 2. Następna mapa na rys. 3 przedstawia ukształtowanie terenu Brazylii. Jeśli interesują nas krainy geograficzne, to kolejna mapa znajduje się na rys. 4. Dodatkowe informacje dotyczące danego kraju znajdziemy w bazie danych. Tabelka zamieszczona na rys. 5 zawiera dane o 10 największych miastach Brazylii, z podaniem liczby mieszkańców kierunkowych numerów telefonicznych, stref czasowych i dokładnego położenia geograficznego. To ostatnie jest niezwykle cenne przy sporządzaniu precyzyjnych horoskopów. Analizując tabelkę na rysunku 6, dowiemy się, że językiem urzędowym Brazylii jest portugalski, ale i angielski jest tam znany. Dalsze informacje będą użyteczne, jeśli zamierzamy spędzić tam urlop (najlepiej w czasie karnawału).

Na rys. 7 znajdują się z kolei pewne dane demograficzne — rozkład wiekowy populacji. Mają one dość regularny układ. Podobne dane dla Polski są bardziej zniekształcone, głównie z powodu I i II wojny światowej. Widać dysproporcje między liczbą kobiet i mężczyzn w pewnych grupach wiekowych.

Dane zawarte w programie mogą być prezentowane także w bardziej globalny sposób. Wykres na rys. 8 przedstawia 10 najbogatszych państw świata. Z kolei mapa na rys. 9 zawiera dane na temat liczby mieszkańców przypadających na jednego lekarza w różnych krajach świata. Środkowa część Afryki to miejsce o najgorszym standardzie opieki medycznej.

Oprócz przeglądania map i danych statystycznych program oferuje również szereg interesujących możliwości: sporządzanie raportów, aktualizację informacji, przeliczenia dewizowe, wykaz stref czasowych, podawanie odległości między dwoma dowolnymi punktami kuli ziemskiej itp.

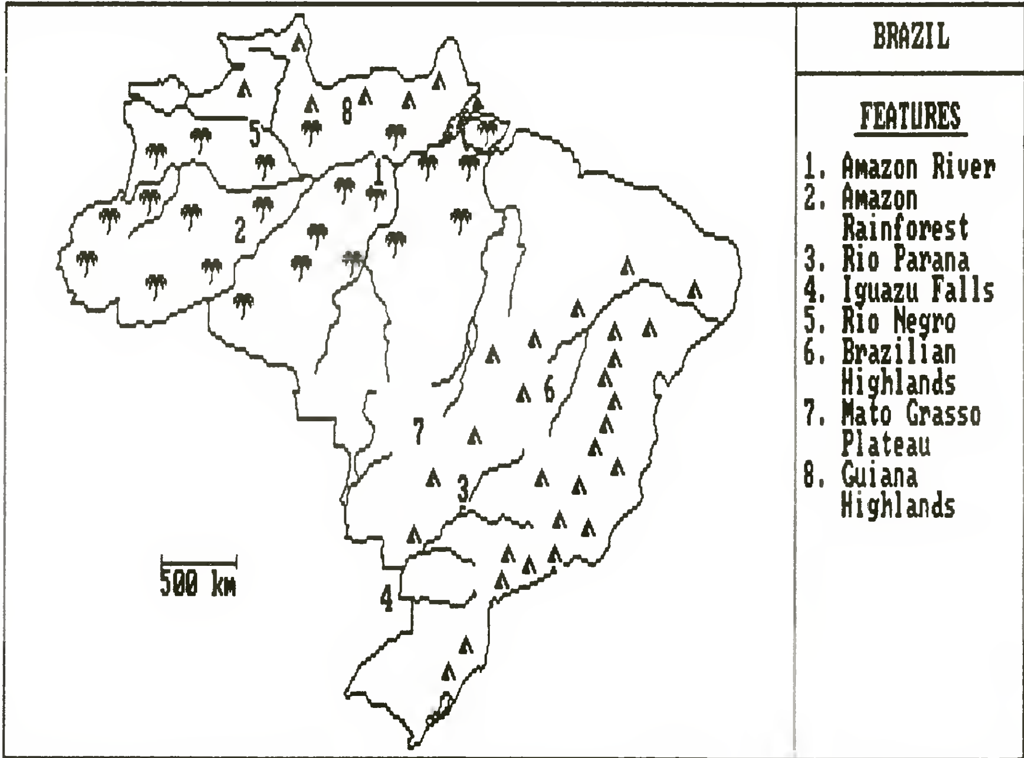
PC USA

Jeśli pasjonowała nas wojna secesyjna, oglądana w serialu „Północ Południe", to z przyjemnością sięgniemy po bliźniaczy program PC USA, którego jeden z ekranów przedstawiamy na rys. 10. Oprócz informacji historycznych możemy dowiedzieć się, jaki stan zajmuje pierwsze miejsce w liczbie rozwodów na 1000 mieszkańców, gdzie produkują najwięcej samochodów, a gdzie są największe zasoby uranu lub innych bogactw naturalnych. Samo spojrzenie na mapę zawierającą skrótowe nazwy Stanów może dostarczyć nam pewnych niespodzia-

Rys. 1. Menu główne programu PC Globe

Rys. 2. Wybrany kontynent — Ameryka Pd.

Rys. 3. Wybrane państwo — Brazylia (ukształtowanie terenu)



Rys. 4. Brazylia — krainy geograficzne

BRAZIL

Major Cities	Population	Internat'l Phone Code	Time Zone	Latitude, Longitude
BRASILIA	1,600,000	55-61	GMT-3	15.45S, 47.55W
Sao Paulo	10,100,000	55-11	GMT-3	23.33S, 46.39W
Rio de Janeiro	5,700,000	55-21	GMT-3	22.53S, 43.17W
Belo Horizonte	2,200,000	55-31	GMT-3	19.54S, 43.54W
Salvador	1,850,000	55-71	GMT-3	12.58S, 38.29W
Fortaleza	1,600,000	55-85	GMT-3	3.45S, 38.35W
Nova Iguacu	1,350,000	55-	GMT-3	22.45S, 43.28W
Recife	1,295,000	55-81	GMT-3	8.06S, 34.53W
Curitiba	1,290,000	55-41	GMT-3	25.25S, 49.25W
Porto Alegre	1,280,000	55-512	GMT-3	30.03S, 51.10W
Belem	1,225,000	55-91	GMT-3	1.27S, 48.29W

- Telex Access Code(s) (Western Union): 391
- Ham Radio Prefix(es): PY

Rys. 5. Brazylia — dane o głównych miastach

BRAZIL

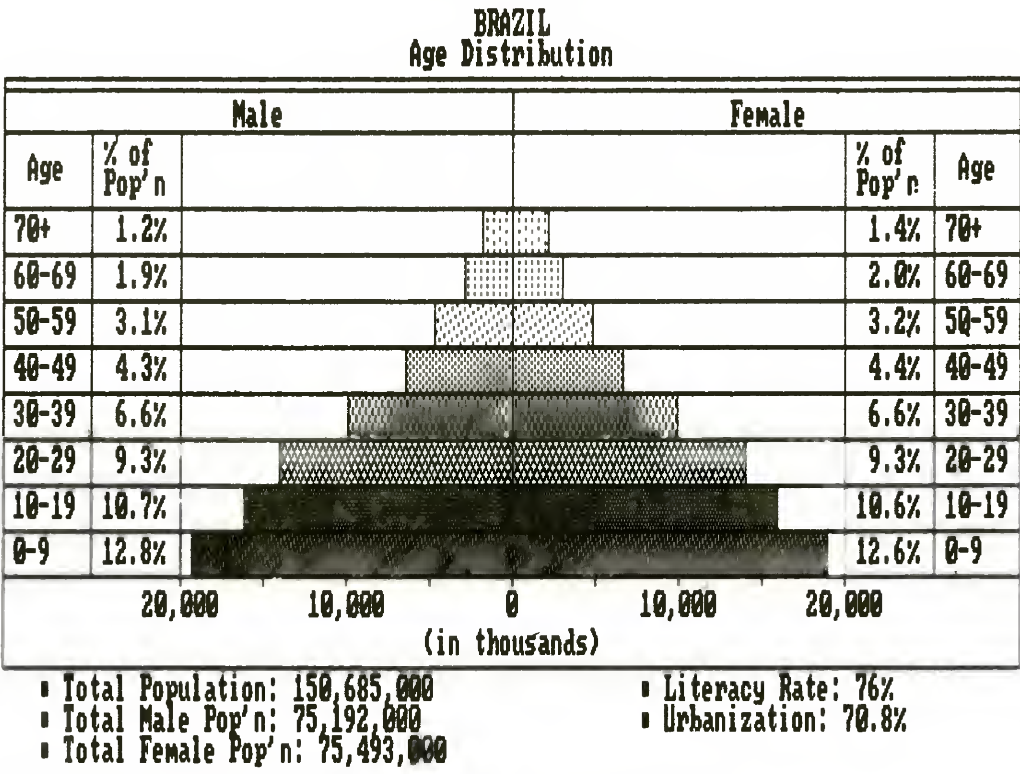
Culture & Tourism

- Official Language: Portuguese. English widely spoken.
- Civilian government re-established in 1985.
- Visa: Required. Return ticket or letter from travel agent assuring onward passage required with application.
- Health: Tapwater not potable. Yellow fever, rabies, hepatitis, typhoid and polio shots recommended.
- Sights: Rio de Janeiro (esp. Carnival), Brasilia, the Iguacu Falls (7th largest in world), the Amazon.
- Climate: Tropical/semitropical. Spring and summer clothes generally suitable.

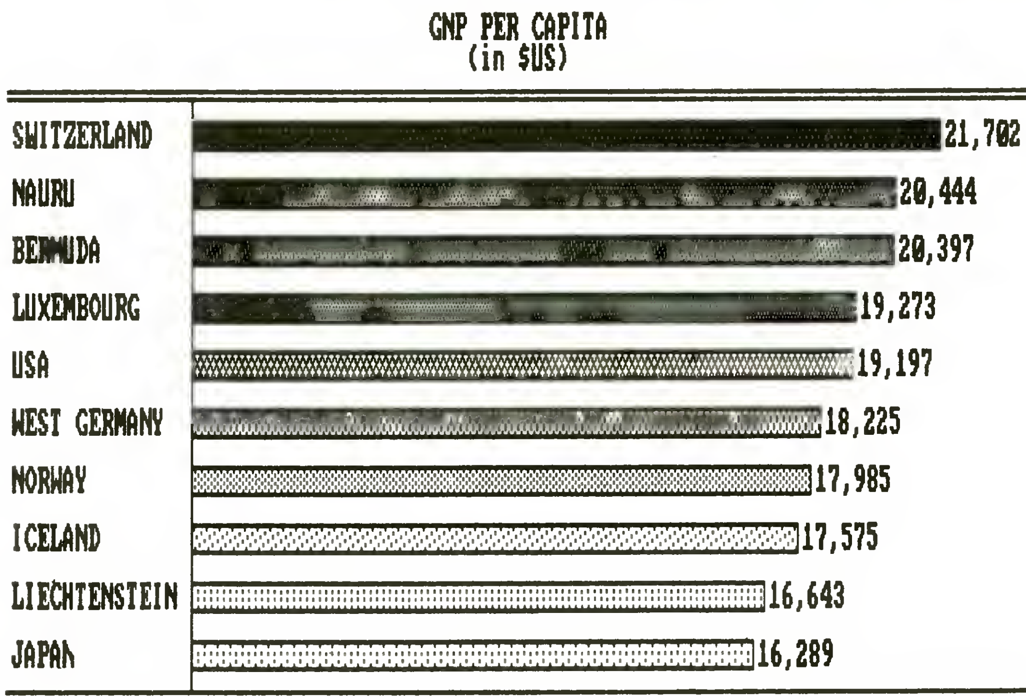
- Annual Tourist Arrivals (1986): 1,934,000
- Annual Tourist Receipts (1986): $1,527,000,000

NOTE: Visa and health requirements are subject to change. Please consult your travel agent or local consulate.

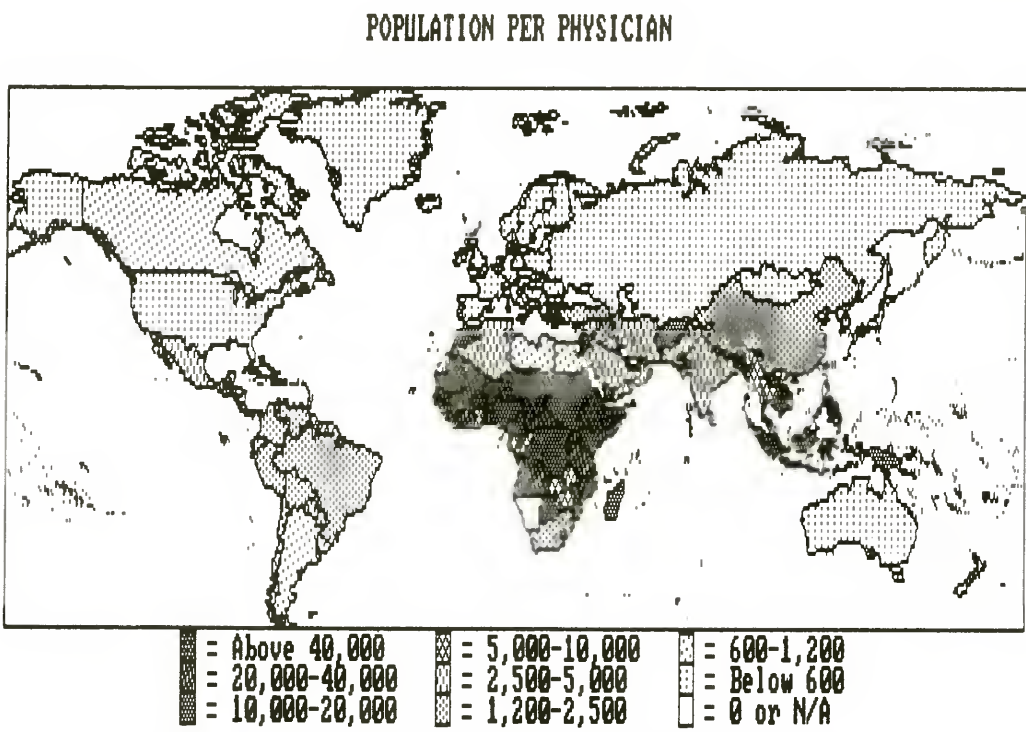
Rys. 6. Brazylia — dane o znaczeniu turystycznym



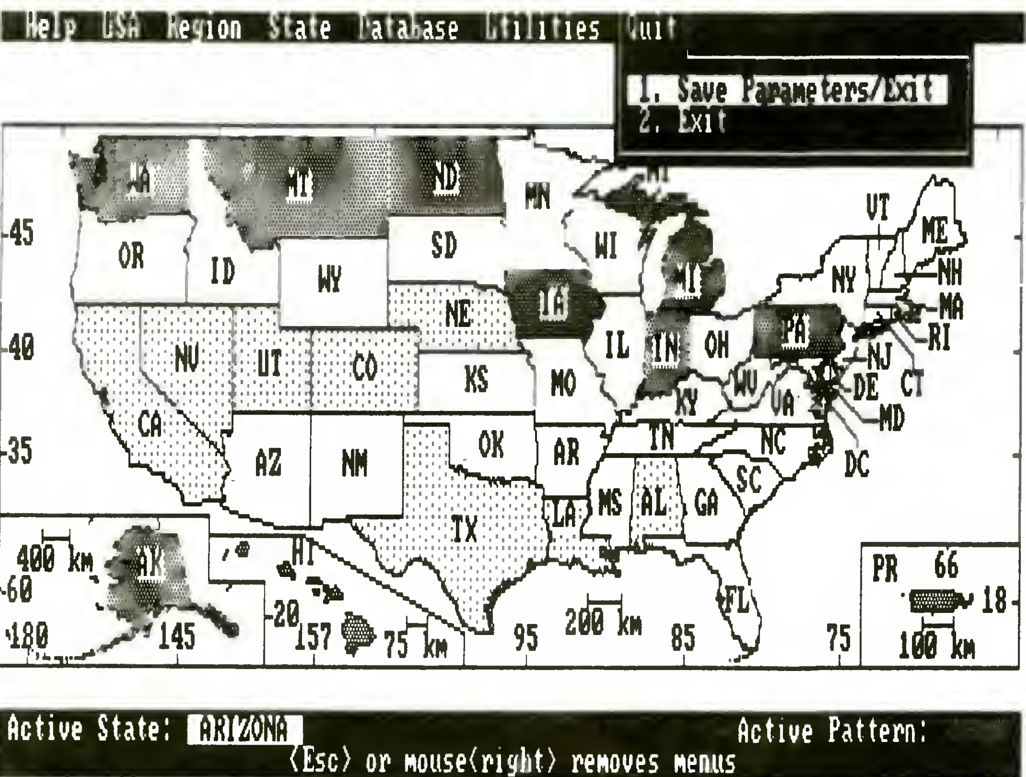
Rys. 7. Brazylia — wiekowy rozkład populacji



Rys. 8. Dane ogólnoświatowe — 10 najbogatszych państw



Rys. 9. Dane ogólnoświatowe — liczba mieszkańców przypadająca na 1 lekarza



Rys. 10. Menu główne programu PC USA

nek. Okaże się, że AL to nie Alaska, lecz Alabama, MO to Missouri, a nie Montana, a miasto Washington nie leży w stanie o tej nazwie.

Wymagania sprzętowe

Oba programy zostały napisane na komputery klasy IMB PC. Współpracują z każdą kartą graficzną, ale bardziej spektakularne efekty zaczynają się od karty EGA i dobrego monitora kolorowego. Duże rozmiary programu i jego baz danych są przyczyną, dla której dopiero przy korzystaniu z twardego dysku mamy znośny komfort pracy. Zalecana jest też kolorowa drukarka i mysz. Minimum pamięci operacyjnej wynosi 512 KB. Aczkolwiek algorytmy rysujące mapy są dość szybkie, to jednak zwykłe XT może być lekko denerwujące.

Podsumowanie

Program można polecić przede wszystkim centralom handlu zagranicznego, urzędom Ministerstwa Spraw Zagranicznych, biurom podróży, dziennikarzom i literatom, a także uczniom i studentom. Oprócz tego sądzę, że wiele osób, które nie przepadały w szkole za geografią, chętnie uzupełni swą wiedzę w tej dziedzinie, posługując się opisywanym programem.

Wszystkim zainteresowanym podajemy adres producenta: PC Globe, Inc., 4435 South Rural Road, Building 5, Suite 333, Tempe, AZ 85282, USA.

Jonasz Mayer

Drogi Bajtku!

Na listy Czytelników odpowiadają autorzy „Bajtka”

1. Proszę podać nazwy programów mających możliwość kopiowania z kasety na dyskietkę.
2. Czym różni się MAC/65 od asemblera?
3. Czy możliwe jest uzyskanie płynnego ruchu ekranu w dół w trybie 15?

Tomasz Machalski
Kraków

1. Programy w Basicu można skopiować wczytując je instrukcją CLOAD i zapisując SAVE„D:nazwa". Kopiowanie z kasety na dyskietkę programów w języku maszynowym wymaga zmiany nagłówka programu. Nie wykonuje tego prawie żaden program kopiujący. Są jednak jeszcze inne sposoby. Programy z „wykrzyknikiem" można skopiować dowolnym programem „np. „FCopy") pomijając przy tym ich pierwszy blok. Niektóre programy można przystosować do pracy z dyskietki za pomocą „Transdisk" lub „Multi-boot". Ostatnio pojawia się ponadto wiele nowych programów kopiujących, lecz trudno podać mi szczegółowe informacje, gdyż nie korzystam z takich programów.

2. MAC/65 jest nazwą własną asemblera, czyli programu służącego do zamiany języka symbolicznego na liczbowy kod maszynowy. Krótko mówiąc, MAC/65 różni się od asemblera tym, czym Atari 65XE różni się od komputera.

3. Płynny ruch ekranu w dowolnym kierunku można uzyskać w każdym trybie graficznym. Jest to stosunkowo proste, lecz wymaga znacznego nakładu pracy.

Podczas eksploatacji stacji dysków LDW Super 2000 odkryłem sposób zmiany gęstości pracy stacji (nie wymieniony w instrukcji). W celu zmiany gęstości należy nacisnąć jednocześnie przyciski ‹DRIVE TYPE› i ‹TRACK›,
Mam też pytanie. Napisałem wyrażenie:
? 2*2=4
po wciśnięciu ‹RETURN› komputer odpowiedział „1". Gdy napisałem:
? 2*2=5
komputer odpowiedział „0". Czy liczby „1" i „0" są zawartością komórki sprawdzającej poprawność działania?

Tomasz Banachowicz
Gdynia

Podczas wykonywania instrukcji PRINT komputer oblicza znajdujące się w niej wyrażenia. Znaki równości i nierówności są przy tym traktowane jako operatory relacji, czyli określają warunek, który musi być sprawdzony. Otrzymujemy więc odpowiednio 4=4 lub 4=5. Pierwszy warunek jest prawdziwy, a drugi fałszywy. Określeniu „prawda" komputer przypisuje wartość 1, „fałsz" zaś ma wartość 0 i te właśnie liczby są ostatecznie wyświetlane na ekranie. Nie jest to więc zawartość jakiejś komórki, lecz wynik wykonania operacji porównania (relacji).

Czy można zmienić procedurę ROM-RAM tak, aby działała w Turbo Basicu? Uważam, że przydałoby się w „Bajtku" więcej artykułów omawiających ten język.

Janusz Krawiec
Grudziądz

Interpreter Turbo Basic XL jest wczytywany w obszar RAM leżący pod ROM systemu operacyjnego, czyli zajmuje dokładnie to samo miejsce, które jest wykorzystywane przez procedurę ROMRAM. Z tego względu nie można ich stosować jednocześnie. Szczegółowy opis interpretera i kompilatora Turbo Basic znajduje się w książce „Języki Atari XL/XE", przygotowywanej do druku przez SOETO.

1. Czy jest możliwość uzyskania większej liczby kolorów w jednej komórce pamięci ekranu?
2. Czy po przełączeniu bloku 16 KB w Atari 130XE traci się jego zawartość?
3. Czy można podzielić punkt w trybie 15 na dwa tak, aby uzyskać kwadrat, a nie prostokąt?
4. Czy jest możliwe uzyskanie w jednej linii trybu 8 kilku kolorów bez stosowania przerwań?

Bartłomiej Marek
Rajcza

1. Punkt ekranu może mieć tylko jeden kolor. Jedna komórka pamięci obrazu może natomiast zawierać informacje o 2—8 punktach obrazu i każdy z nich może mieć jeden z kolorów dozwolonych w danym trybie.

2. Odłączenie bloku uniemożliwia procesowi dostęp do niego, układ pamięci jest jednak nadal zasilany prądem i zawartość nie jest tracona.

3. W żadnym trybie graficznym nie ma możliwości podzielenia punktu na mniejsze części. Można tylko zmienić tryb. Gdyby jednak (teoretycznie) udało się podzielić punkt w trybie 15, to otrzyma się dwa punkty w trybie 8.

4. Teoretycznie nie. W praktyce kolor punktu w trybie 8 zależy od jego położenia w parzystej lub nieparzystej kolumnie ekranu oraz od koloru sąsiednich punktów.

Ograniczenie wskazane w punktach 1 i 3 oraz efekt opisany w punkcie 4 wynikają z konstrukcji kineskopu w monitorze lub telewizorze oraz z ustalonej przez projektantów komputera metody współpracy układu ANTIC z monitorem. Programowo nie można tego zmienić. Zwiększenie rozdzielczości i liczby kolorów wymaga zmiany komputera (czasem również monitora).

Interesuję się samochodami. Od dwóch lat mam Commodore 64. Staram się zdobywać jak najwięcej gier samochodowych. (...) Nie spotkałem jednak dotąd gry idealnej. Po przeczytaniu w „Bajtku" opisu gry Grand Prix Circuit wiedziałem, że to ona! Po pewnym czasie mój entuzjazm ustąpił miejsca rozczarowaniu. NIKT nie miał tej gry. Co mam robić? Jesteście moją jedyną szansą.

Mirosław Rus, ul. Szpitalna 18/6
41-600 Świętochłowice

Problem rzeczywiście jest poważny. Uczucie bezsilności w poszukiwaniu wymarzonej gry powoduje bowiem ból większy niż po stracie ukochanej. My mieliśmy Grand Prix Circuit, o czym świadczy zdjęcie. Dawno już ją jednak skasowaliśmy i w zasięgu nie ma nikogo, kto posiadałby ją. Za to często gramy w nią na IBM i jest niezgorsza. Co do wersji na Commodore, to zajmuje ona całą stronę dysku, więc jeśli masz magnetofon, to mam wątpliwości, czy zagrasz...

Czym różni się stacja dysków FDD 3000 do FDD 3?

Stefan Wichniewicz
Zamość

Stacja FDD 3000 to produkt portugalskiej filii Timexa, natomiast FDD 3 — to podróbka rodzimego Polbritu. Przede wszystkim różnią je gabaryty oraz pojemność pamięci. FDD 3000 jest duża, poprzeczna i posiada miejsce na dwa 3" napędy. Natomiast FDD 3 jest podłużna, niewielka i z miejscem na jeden napęd. Pierwsza posiada 64 K RAM, druga zaś tylko 16 K. Tym samym niemożliwe jest uruchomienie CP/M-u na FDD 3 ani korzystanie ze skomplikowanych programów wykorzystujących pamięć stacji. Możliwa jest przeróbka, czego już dokonaliśmy. Efekt w postaci schematu znajdziecie wkrótce w „Bajtku".

(...) Dlaczego program w kodzie maszynowym wysyłający na ekran znaki procedurą RST 16 działa w ten sposób, że wypisuje je w oknie systemowym i komunikuje Out of screen?

Mariusz Krzyżanowski
Lublin

Chodzi zapewne o progam

```
LD A,65   ; kod znaku
RST 16    ; wysłanie kanałem
```

Litera „A" zostanie wysłana otwartym kanałem. Ponieważ w trybie edycji otwarty jest kanał 0 (okno systemowe), tam pojawi się litera. Jeśli będzie ich więcej, interpreter zakomunikuje wyjście za ekran.

Lekarstwem jest otwarcie kanału 2 (ekran). Teraz program będzie wyglądał tak:

```
LD A,2      ; numer kanału
CALL 5633   ; otwarcie kanału
LD A,65     ; kod „A"
RST 16      ; wydruk
```

Zmieniając 2 na 3 otrzymalibyśmy wydruk na drukarce.

Istnieje jeszcze jeden sposób, nieco oszukańczy. Należy pośrednio otworzyć kanał na ekran (automatycznie robi to np. PRINT) wykonując PRINT AT 0.0;: RANDOMIZE USR adres, jeśli pod adresem „adres" znajduje się podany na początku programik.

UWAGA! POSIADACZE 8-MIO BITOWEGO ATARI!
- SPARTA DOS X -
- REWELACYJNY DYSKOWY SYSTEM OPERACYJNY W PAMIĘCI ROM -
PRODUKOWANY NA LICENCJI FIRMY ICD INC.

- operuje standardowym zestawem komend stosowanym przez DOS'y 16-to bitowych komputerów (MS DOS, PC DOS);
- nie zajmuje miejsca w pamięci RAM komputera (memlo ‹$1000);
- zapewnia szybką transmisję danych przy współpracy ze stacjami US DOUBLER, HAPPY, INDUS, LDW 2000, CA 2002, XF551;
- współpracuje ze stacjami dwustronnymi, 720 KB i twardym dyskiem;
- nie koliduje z TURBO BASIC'em XE, ATARIWRITER+ i większością programów dla komputerów ATARI na dyskietkach i cartridge'ach;
- odczytuje dyskietki zapisane przez inne DOS'y (DOS 2.x, DOS 3.x, MY DOS 4.5);

SPARTA DOS X - TO REWELACYJNE NARZĘDZIE PRACY ZE STACJĄ DYSKÓW
DLA KAŻDEGO, KTO KOMPUTERA UŻYWA NIE TYLKO DO GIER !
INFORMACJE I SPRZEDAŻ: PZ KAREN

05-070 SULEJÓWEK
AL. MARSZ. PIŁSUDSKIEGO 2,
Tel.715-42 wew.449, telex 813 948 kren pl

02-361 WARSZAWA
UL PRZEMYSKA 11A
TEL 659-34-14

Ogłoszenia drobne

Mikroservice Commodore-64/128--Amiga, Spectrum-PC/XT/AT Atari CARTRIDGE 01—911 Warszawa, Andersena 3/103. Bd 3

COMMODORE 64- „Nieśmiertelność" do gier taśmowych i dyskietkowych. A. Urbankowski, ul. Czarnogórska 8/7, 30—681 Kraków. Bd 8

Programy ATARI Nowości — ul. W. Skotnica 20/40, 41—400 Mysłowice. Bd 34

GEODECI! Programy na Atari XL/XE, Zgłoszenia: Mieczysław Kozłowski 19—200 Grajewo, Os. Waltera 25/44, tel. 31—45. Bd 32

TANIO! Commodore 64-programy, catalog gratis, wysyłka na kraj Studio „Ω" Dąbrówka-Stany 62. 08—114 Skórzec. Bd 29

Rewelacyjne niskie ceny, najkrótsze terminy, najwyższa jakość!!! Programy na Atari XL/XE. Aler i Robert ul. Niecała nr 9 36—100 Kolbuszowa. (informacja–koperta+znaczek). Bd 28

ATARI XL/XE. Szeroki wybór gier i programów użytkowych. Laskowski Grzegorz, 07—412 Ostrołęka, ul. Jaworskiego 5/15. (Informacje koperta + znaczek). Bd 27

„Spectrum. Nowości na taśmie lub dysku. RAJSOFT, Kilińskiego 27/17 96—300 Żyrardów" Bd 26

Spectrum, Timex, najnowsze programy, Robert Gruzielewicz ul. Działkowa 15, 96—300 Żyrardów. Bd 25

Sprzedam układy redukcji szumów do magnetofonów Migh Com i dbx. Informacje–koperta+znaczek. Inż. Andrzej Ściślicki 42—450 Łazy skr. poczt. 58. Bd 23

Gry komputerowe na ATARI, SPECTRUM, AMSTRAD. Nagrywanie na kasety i dyskietki. Toruń, Moniuszki 16/20, pok. 1 Godz. 12.00—14.00 Tel. 247—31, 224—13. Bd 17

Mieszkańcy wsi i miasteczek. SDS Atari-programy tanio, szybko (koperta + znaczek) Opole, Krajewskiego 19A/603. 458 K

ISR

International Service Resources

Warszawa
ul. Okrężna 3
tel. 42 44 30
fax 42 44 30





- jedyny autoryzowany w Polsce serwis gwarancyjny i pogwarancyjny sprzętu Commodore
- urządzenia peryferyjne i oprogramowanie do Commodore 64 i Amigi
- u nas kupisz Amigę 2000, najtańszy na świecie w pełni profesjonalny komputer o olbrzymich możliwościach graficznych, animacyjnych i muzycznych. Dla grafika, inżyniera, firmy video, agencji reklamowej polecamy także komputery Commodore serii PC
- instalujemy licencjonowane sieci Novell w oparciu o sprzęt renomowanych firm amerykańskich: Everex, Maxum, Tandy.

Wojewódzkie Przedsiębiorstwo Handlu Wewnętrznego Oddział w Tychach

VIDEOBIT

43-100 Tychy, Al.ZMP 77
tel.276975

poleca między innymi

— sprzęt komputerowy
Atari ● Commodore ● Amstrad
● IBM PC XT/AT/PS 2 ●
— drukarki STAR, EPSON, AMSTRAD
— Sprzęt audiowizualny
— magnetowidy
— OTV PAL/SECAM
— Videoskopy
— kamery
— anteny satelitarne
— aparaturę badawczo-naukową

Udzielamy gwarancji, prowadzimy naprawy pogwarancyjne. Zapewniamy o atrakcyjnych cenach.

B 7

MICROMAN

● ● ● ● ● ● ● ●

oferuje:
1. Programy i literaturę dla komputerów:
Atari XL/XE/ST, Commodore 16/116/+4/64/128/Amiga, Spectrum, Timex na miejscu lub za zaliczeniem pocztowym.
2. Dla Atari XL/XE oprogramowanie na cartridge'u.
3. Przystawki „UNIWERSAL TURBO" dla magnetofonów firmowych, umożliwiające zapis i odczyt programów zarówno w systemie Blizzard jak i Turbo 2000.
4. Naprawy zasilaczy, magnetofonów, klawiatur w komputerach Atari, Commodore, Spectrum, meritum.
5. Dla użytkowników Elwro 800 program kopiujący taśma-dysk.
6. Dodatkowe akcesoria dla komputerów domowych.
Ponadto wykonuje i naprawia nietypowe urządzenia elektroniczne. Informacje na miejscu lub za załączeniem koperty zwrotnej.
Adres: **MICROMAN**
40—181 Katowice
ul. Osikowa 66
tel. 585—106

B 6

KOMPUTER NATYCHMIAST
KUPISZ-SPRZEDASZ

MAXSOFT

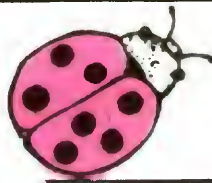
659-44-17 Warszawa

(SB-87)

ATARIX

ATARI 800 XL
65XE 130XE
literatura
oprogramowanie
katalogi gratis
nizawodność
Warszawa 04-357
Grochowska 186 m 69

B51



ZAKŁAD ELEKTRONICZNY

ATRAX

oferuje dla odbiorców indywidualnych i hurtowych najtańszy, niezawodny osprzęt i urządzenia peryferyjne do komputerów domowych:

COMMODORE 64/128

- cartridge: X, Black Box, Final II, Final III, Action Replay
- interface drukarki typu Centronics

AMIGA 500

- rozszerzenie pamięci o 0,5Mb z zegarem lub bez
- stacja 5 1/4 cala

ATARI XL i XE

- interface magnetofonu
- interface drukarki Centronics
- cartridge: assembler-editor, Logo, Basic XE, Basic XL action, Turbo 2000 K.S.O, Turbo 2000F, Turbo 2000 Copy,

SPARTA DOS

- moduł Turbo do montażu w magnetofonie Atari
- top drive stacji 1050

ATARI ST

- stacja 5 1/4 cala
- cartridge Multiface (kopiowanie zabezpieczonych programów, organizacja dysku itp)

Możliwość wykonania urządzeń na indywidualne zamówienie. Szczegółowe informacje wysyłamy pocztą po otrzymaniu koperty ze znaczkiem.
ul. Biedronki 83
02—949 Warszawa, Wilanów

B 39

Serwis Komputerów

TEST

Katowice, ul. Armi Czerwonej 22/53 tel. 598322 (superjednostka) IX piętro
poleca naprawy:
- ATARI 600, 800, 65, 130 XL, XE
- COMMODORE 16, 116,+4, 64, 128, 1280, AMIGA
- DISK DRIVE 1541, 1570, 1571, 1050

rozszerzenie pamięci:
- ATARI 600XL, COMMODORE 16, 116, do 64kb
- ATARI 800XL, 65 XE, do 130 kB
- AMIGA 500 do 1 MB

godz.9–11,15–18

SB 30

KOMPUTER — SERVICE
Naprawa komputerów
COMMODORE, IBM, SPECTRUM, TIMEX
oraz
SERWIS i przeróbki
zasilacze monitorów, drukarek.
Instalacje polskich znaków.
Kraków, tel. (012)33-96-51
poniedziałek — piątek
godz. 10.00 – 13.00,
20.00 – 21.00

B 15

SAM WYKONASZ OBWODY DRUKOWANE

Zestaw (laminat, odczynniki, instrukcja)
Cena 3550 zł. plus opłaty pocztowe. Wysyłka za zaliczeniem pocztowym. Zamówienia kierować: A. Kawczyński 90–950 Łódź-1 skrytka pocztowa 344. Płatne przy odbiorze paczki. ZAWSZE AKTUALNE!

B 11

Studio Komputerowe

ATARMAN

oferuje oprogramowanie dla komputerów:
ATARI XL/XE, COMMODORE 64/128, TIMEX, SPECTRUM, AMIGA, MSX
— gry
— programy użytkowe
— programy demonstracyjne
— wszystkie nowości
— programy niezabezpieczone
— co piąty program gratis
— pomoc dla początkujących
— wysyłka pocztą na cały kraj
— katalogi gratis
— przeróbki sprzętowe

Adres: ul. Niepodległości 24/9
20-246 Lublin
tel. 77-22-51

NAPISZ!

B 45

Klawiatury z kontakturą do instrumentów muzycznych poleca:
Stanisław Putyra 43-400 Cieszyn, Szymanowskiego 3c/40.
Informacja — opłacona koperta zwrotna.

Bd 41

Atari. Najnowsze programy — Blizzard i normal. 41-400 Mysłowice ul. Skotnicą 32/30

Bd 48

ATARI XL/XE
programy na dyskach (sprzedaż, wymiana), zapisywanie, kopiowanie EPROM-ów
opisy programów, sprzętu
Informacje, katalogi — bezpłatnie
Wiesław Faruga, Pogórze 133, 43-430 Skoczów

B 42

„Naprawa komputerów i klawiatur ZX SPECTRUM Zabrze-Helenka, ul. Jaworowa 6/2 w godz. 17.00–20.00."

B 24

ATARI ST.STE
520 — 1 MEGA

Gry, programy użytkowe.
NOWOŚĆ !!! Katalog gier na kasecie VIDEO.
Unikalne instrukcje! Falcon, Gun Ship, Bard's Tale, Populous, North & South, itp...

ATARI XL/XE

Wszystkie programy kasetowe sprawdzane po nagraniu!!!
STUDIO KOMPUTEROWE
„HEGATAR"

HALA WOLA
CZŁUCHOWSKA 25

godz. 16–19,
01-360 WARSZAWA

B 13

wszystko o...

ATARI ST

Najlepsze programy, instrukcje, literatura, własne unikalne opracowania, porady.
UWAGA! NOWOŚĆ — Pierwszy w Polsce katalog programów na kasecie wideo!
KOMPUTER STUDIO Warszawa al. Marsa 6 tel. 15-42-20
godz. 14–19

B 21

Naprawa komputerów SPECTRUM, COMMODORE, TIMEX

oraz monitorów, drukarek, zasilaczy.

Programowanie EPROMów.

30-681 Kraków 47
skr. poczt. 75, tel. (012)44-79-26
poniedziałek – środa
w godz. 9.00 – 11.00

B 33

UWAGA!

Jedyny w Polsce listowny Klub Komputerowy
COMMODORE 64, 128
zrzeszający członków z kraju a także z zagranicy udostępniający gry i programy
DARMO!
Napisz a zostaną udzielone Ci informacje. 59-300 Lublin ul. Zawadzkiego 50/7

B 19

ATARI

programy pocztą
„HOBBIT" Sp. z o.o.
22-600 Tomaszów Lubelski, ul. Kraszewskiego 12 tel. 28-39
Oferuje: szeroki wybór gier, programów użytkowych i instrukcji. Szczegółowe informacje i katalog gratis po załączeniu koperty i znaczka.
Co piąty program gratis,
programów nie zabezpieczamy.

B 36

COMMODORE
ZX SPECTRUM

najnowsze gry

NISKIE CENY

katalogi gratis
skr. poczt. 5
07-200 Wyszków

B 22

Atari Turbo
2000 F:

Nowy system transmisji danych z magnetofonem przyspieszony do 6700 bodów.
Komplet:
— cartridge
— oprogramowanie
— przeróbka magnetofonu
— instrukcja obsługi
— 12 miesięcy gwarancji.
Instalacje wykonujemy na poczekaniu.

Interfejs do zwykłego magnetofonu.
Duży wybór oprogramowania w standardzie TURBO-2000.
Informacja:
Tel. 33-40-91
Korespondencja i wykonywanie usług:
MUEL ul. Cząstkowska 30
01-678 Warszawa.

B 31

ATARI 800 XL,
65 XE, 130 XE

Sprzedaż wysyłkowa gier i programów użytkowych na kasetach i dyskietkach.
Również w systemie TURBO 2000
Wszystkie nowości!!!
Instrukcje i literatura.
Dla zainteresowanych rachunki.

ANWIKOL

03-721 Warszawa ul. Jagielońska 3/28.

B 10

DŁUGOŚĆ ŻYCIA.

Program oparty na długoletnich badaniach naukowych. Atari zadając szereg pytań oblicza długość Twojego życia. Cena programu wraz z kasetą wynosi 15.000,- + +koszty przesyłki. Tomasz Nowak, Sadków 23, 55-080 Kąty wr.

B 37

ATARI
TURBO-2000

do samodzielnego montażu. Szczegółowe informacje po nadesłaniu zaadresowanej koperty ze znaczkiem na adres: ul. Kilińskiego 47A/2 82-300 ELBLĄG

B 20

Elementy Elektroniczne — Skup i Sprzedaż szeroki asortyment — umiarkowane ceny Przyjmę zamówienia od osób prywatnych i instytucji (rychunki). Prowadzę sprzedaż wysyłkową.
Oferta — koperta zwrotna + znaczek.
Lesław Buras, 51-639 Wrocław ul. Wyczołkowskiego 17

B 50

INFORMATYKA W SZKOLE VI

Błażejewko k/Kórnika, 17–20 września 1990

W dniach 17–20 września 1990 roku odbędzie się w Błażejewku k/Kórnika kolejna, VI Konferencja **INFORMATYKA W SZKOLE**. Tegoroczna Konferencja będzie poświęcona przede wszystkim prezentacji oprogramowania wspomagającego nauczanie w szkołach podstawowych i średnich. Uczestników upatruje się w nauczycielach szkół podstawowych, średnich i wyższych, wizytatorach i metodykach, oraz pracownikach naukowo-badawczych. Do udziału w Konferencji zostali zaproszeni m.in. wykonawcy systemów oprogramowania w programach RPI.09, RRI.14 i RRI.16. Zapraszamy także firmy zajmujące się wytwarzaniem i dystrybucją sprzętu i oprogramowania dydaktycznego.

Organizatorami Konferencji są: Instytut Informatyki Uniwersytetu Wrocławskiego, Ministerstwo Edukacji Narodowej i Ogólnopolska Fundacja Edukacji Komputerowej — Oddział Wlkp.

Zgłoszenia udziału w Konferencji i oprogramowania do prezentacji należy nadsyłać na adres: Instytut Informatyki (Informatyka w Szkole VI), Uniwersytet Wrocławski, ul. Przesmyckiego 20, 51151 Wrocław.

Koszt udziału w Konferencji wynosi: **350 000 zł**.

Maciej M. Sysło (IInfUWr)

B 35

SUPER OFERTA — ATARI XL/XE — SUPER KATALOG

★ największy wybór opisów do gier i programów użytkowych: literatura
★ gry, programy narzędziowe, użytkowe, polskie programy edukacyjne
★ interfejsy do magnetofonów w systemie standardowym, ATARI SUPER TURBO, TURBO 2000, przeróbki magnetofonów firmowych
★ TOP DRIVE — samodzielny montaż
★ interfejsy CENTRONICS do drukarek

COMMODORE

★ kartridże X, FINAL II, FINAL III

SPRZEDAŻ WYSYŁKOWA — GWARANCJA

STUDIO KOMPUTEROWE MEGABAJT
03-945 WARSZAWA 33
skr. poczt. 28. tel. 17-76-16

B 14

ATASERW

43-100 TYCHY
ul. Lencewicza 46/3
tel. 27 69 66

oferuje świetne rozwiązania sprzętowe

do ATARI XL/XE:

1. TURBO DOS — wspaniały DOS na kartridżu
2. TOP DRIVE — do stacji 1050, LDW 2000, CALIFORNIA samodzielny montaż — (rec. INFORMIK III/88)
3. INTERFEJS CENTRONIKS
4. ROZSZERZENIA PAMIĘCI
5. BASIC XE — kartridż
6. TUR DOS+BUG65+MAC65 —kartridż

12 miesięcy gwarancji. Informacje i zamówienia telefoniczne (wtorek 8–12, środa, czwartek (16–18) i listownie po otrzymaniu koperty zwrotnej.

B 16

WPPUH spółka z o.o.

„ATARES"

Chorzów ul. Truchana 35, tel. 500–797 w godzinach od 18–21 oferuje użytkownikom **ATARI** najnowsze rozwiązania sprzętowe podnoszące komfort pracy z komputerem m.in.:

- BLIZZARD TURBO — kasetowy system transmisji DANYCH 6000 bodów (magnetofon po przeróbce czyta 10 razy szybciej, praktycznie bez błędów)
- CRYSTAL SOUND — system digitalizacji dźwięku
- CARTRIDGE do obsługi systemu BLIZZARD TURBO (6 typów) i inne
- INTERFACE CBT — do magnetofonów niefirmowych (realizuje transmisję standardową i w TURBO BLIZZARD 6000 B)
- INTERFACE STANDARD do magnetofonów niefirmowych (odczyt — zapis 600 Bodów
- INTERFACE CENTRONICS do współpracy z drukarkami i plotterami
- PROGRAMATOR EPROM PE-1
- TURBO DRIVE 1050 (LDW 2000) format dysku 180 kB, transmisja 70 kB
- MAXI TURBO DRIVE 1050 — format dysku 180 kB, transmisja 140 kB
- ROZSZERZENIA PAMIĘCI ATARI do 256 kB włącznie
- SERWIS NAPRAW POGWARANCYJNYCH sprzętu ATARI, COMMODORE, SPECTRUM
- OPROGRAMOWANIE do ATARI, COMMODORE (w tym AMIGA), SPECTRUM

W PRZYGOTOWANIU

TELETEKST na bazie ATARI oraz COMMODORE C-64
HOME PHOTO — przetwornik obrazu ATARI
SUPER SONG — generator dźwięku do C-16, C+4
VIDEO LETTER — system nakładania napisów na obraz video

UDZIELAMY ROCZNEJ GWARANCJI, w przypadku magnetofonów ATARI po montażu TURBO przejmujemy GWARANCJĘ po PEWEX-ie

ZAPRASZAMY PT KLIENTÓW DO WIELOBRANŻOWEGO SKLEPU PRZEMYSŁOWEGO przy ul. Truchana 35 codziennie w godz. 9-17

Prowadzimy również sprzedaż pozarynkową.

B–40

Wszystko do komputerów ATARI



ATARI STUDIO
W-wa ul. Jóźwiaka 4
10°°-18°° ☎ 125-123

ATARI SUPER TURBO
PIERWSZY I NAJLEPSZY SYSTEM TURBO DO ATARI

- Nowe programy narzędziowe do systemu AST
AST Monitor, AST Emulator, Head Test, itd
- Rewelacyjny nowy cartridge z "narzędziami"
- Autoryzowany montaż, serwis, pełna obsługa
- Wszystkie programy w AST, także nowości !

ATARI XL/XE, ST

Polecamy pełny wybór programów, literatury i instrukcji do wszystkich 8-io i 16-to bitowych komputerów ATARI. Zapraszamy do współpracy!

W KAŻDEJ SPRAWIE I CHWILI

B 43

Prosimy o kontakt wszystkich piszących programy na ATARI

ATARI

Montaż udoskonalonego systemu Turbo-Rom-Plus w magnetofonach **ATARI** co najmniej 80 gier z loaderami na kasecie C 60 wczytywanych start + option Fend w 1 min. i 24 sek, również praca w Blizzardzie — licencja programowa Firmy Atares Honorujemy gwarancję firmową.

Zakład Elektroniki „PLUS"

Kraków, ul. Mochnackiego 67
godz. 10–18 sob. 9–13
tlf. 33–23–12

Punkty przyjęć:

Tarnów, ul. Traugutta 7/10, środy 16–18

Rzeszów, ul. Rejtana 43/6 środy 10–14 tel. 548–82

B–46

MÓZGPROCESOR!

to rewelacyjna polska gra przygodowa firmy

COMPUTER ADVENTURE STUDIO dla Atari XL/XE (taśma + opis), dla Spectrum, Timex, Junior (taśma + opis), — dla Atari XL/XE (dyskietka + opis). Cena zestawu — równowartość 2,5 USD. Test w „Bajtku nr 10/89.

Zamówienia prosimy kierować:

COMPUTER ADVENTURE STUDIO

32-700 Bochnia, ul. Kazimierza Wielkiego 37/45
tel. (o-197) 242-47 8–16

UWAGA!
Nawiążemy współpracę z autorami orginalnych polskich programów i scenariuszy.

B 4

Czytelniku!!!

Brakujące numery Bajtka możesz nabyć w naszej redakcji 00-687 Warszawa, ul. Wspólna 61 w godzinach 9.00–16.00

cena 3000 zł za egzemplarz

1987

5, 6, 7, 8, 11, 12

1988

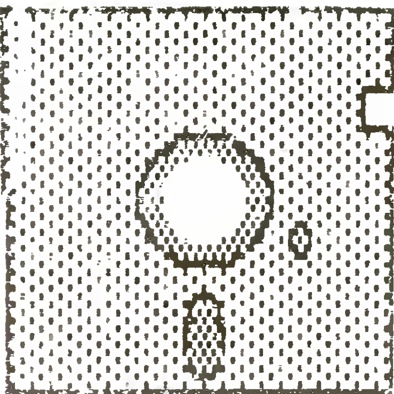
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

1989

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

	Giełda	Sklep	Pewex	Zachód
	tys. zł		$	
SINCLAIR				
ZX 81	—	—	—	—
ZX Spectrum 48	880	950	—	100
ZX Spectrum +	950	1050	—	125
ZX Spectrum + 2	1300	—	—	170
Timex 2048	1000	1200	—	—
stacja FDD3000	800	1450	—	—
stacja FDD3	650	—	—	—
drukarka GP–50	—	480	—	—
Masterface I	100	—	—	—
AY 3–8910	120	—	—	—
COMMODORE				
C 64	1400	1700	199	210
C 64 Desk–Top	1900	—	—	—
C 128	2100	2300	—	250
C 128 D	4300	3900	599	—
Amiga 500	5200	5900	765	700
Amiga 2000	18000	—	—	2000
magnetofon	270	290	30	—
stacja 1541	1500	1700	—	—
stacja 1571	1900	2300	—	220
stacja Oceanic	1500	—	—	120
LC 10C	2300	2500	299	260
Final II	90	—	—	—
Final III	200	—	—	—
Action V	300	—	—	—
ATARI				
800 XL	1000	1100	—	—
65 XE	1100	1150	127	—
130 XE	1400	1600	199	185
520 STFM	4000	4300	499	350
1040 STFM	—	—	899	840
magnetofon	350	350	51	—
stacja 1050	1200	1400	—	130
monitor SM124	2000	—	199	—
monitor SM1224	—	—	479	470
drukarka 1029	800	1300	—	—
Turbo 2000	100	100	—	—
Centronics	180	190	—	—
AMSTRAD				
464	2000	1700	—	—
664	—	2300	—	—
6128	2700	2900	—	320
PCW 8256	—	—	—	—
PCW 8512	—	5200	—	—
PC 1512	10.000	—	—	700
PC 1640	14.000	—	—	1100
IBM				
PC XT stand.	7000	7500	499	550
PC AT stand.	—	—	—	700
HD 20 MB	3000	2700	370	260
napęd 5″	600	—	100	50
monitor amber	—	—	219	140
klawiatura	700	350	—	190
INNE				
dyskietki 5″	3.2–5	4.5–10	—	0.3–2
dyskietki 3.5″	8–13	9–12	2	1–3
dyskietki 3″	19–25	19.5	—	2–4
kasety C–60	12	—	1	1
monitor Neptun	500	650	—	—
joystick	50–80	50–85	7–9	10–25

Sklepy „Bajtka": Bytom, ul. Koniewa 6, tel. (832) 81-57-01
Kraków, ul. Pstrowskiego tel. (012) 56-54-52



INDYWIDUALNY BANK DANYCH

Kolbrecki Władysław, posiada Spectrum +3. Poszukuje programu CP/M. Adres: Ostrobramska 84 m 13, 04-163 Warszawa, tel. 15 61 35.
Przemysław Jeziorski, lat 13 posiada komputer Amstrad CPC 464 wraz z grami w wersji kasetowej. Proponuje wymianę programów i doświadczeń. Adres: 10–284 Olsztyn, ul. Kolejowa 7 m 1, tel. 26 96 34.
Piotr Nowak, posiada Atari 800XE i XC12, około 200 programów. Odpowie na każdy list. Adres: ul. Rejtana 4b/8, 74-200 Pyrzyce.
Paweł Leśniak, lat 15. Posiada Amstrada CPC 6128. Pragnie nawiązać kontakt z posiadaczami tego komputera w celu wymiany doświadczeń i oprogramowania. Adres: os. Wilanów III, ul. Wiktorii Wiedeńskiej 7/10, 02-954 Warszawa, tel. 40 91 74.
Krzysztof Jabłoński, lat 16 posiada Amiga 500 i około 30-stu programów. Interesuje się muzyką i grafiką komputerową. Nawiąże kontakt w celu wymiany oprogramowania, doświadczeń i literatury (może być w j. angielskim). Adres: ul. Wybickiego 8, 51-144 Wrocław, tel. 25 22 49.
Mirosław Wieczorek, lat 39, posiada Atari 130, magnetofon, oraz około 400 programów, głównie gier. Wymieni oprogramowanie. Adres: ul. Kościuszki 9, 57-520 Długopole Zdrój.
Krzysztof Nadowski, lat 37, nawiąże kontakt z użytkownikami komputera NEC APC 3 (procesor INTEL 8086, system MS DOS 2.11). Adres: ul. Niepodległości 4 m 4, 38-300 Gorlice.
Dariusz Barczyński, lat 16, posiada Timex-2048 i około 800 programów. Chętnie wymienię oprogramowanie. Adres: ul. Raszyńska 39/8, 60-135 Poznań.
Jan Formanek, lat 16, posiada Atari 1040 STE z monitorem monochromatycznym SM 124. Prosi o kontakt w celu wymiany oprogramowania i literatury. Adres: ČERNÉHO 517, PRAHA 8 182-00 STRIZKOV.
Daniel Sochacki, lat 16, posiada Commodore 64, magnetofon 1530 Datassette, monitor monochromatyczny, cartridge–"X". Dysponuje zestawem 600 programów. Proponuje wymianę gier i programów, odpisze na każdy list. Adres: ul. Żywiecka 62, 43-365 Wilkowice.
Remigiusz Wagner, lat 15. Komputer Amstrad CPC 464. Nawiąże kontakt z posiadaczami tego komputera. Adres: ul. Fabryczna 51/14 66-400 Gorzów Wlkp.
Łukasz Graczyk, posiada Atari 65 XE poszukuje oprogramowania: Gyrus Ninja, Wrestling, Jumbo Jet Pilot, Agent U.S.A., i inne. Adres: ul. Adama Grzymały Siedleckiego 34/15, 85-868 Bydgoszcz.
Paweł Garczarek, lat 16. Posiada ZX Spectrum + i około 1000 programów. Proponuje wymianę oprogramowania. Adres: ul. Raszyńska 37/38, 60-135 Poznań.

joy

JOYSTICKI do Atari Commodore, Spectrum, Amstrad precyzyjny mechanizm specjalne styki, kable do joysticka z wtyczką 6 m-cy gwarancja interface do Spectrum

Wysyłka natychmiastowa za zaliczeniem pocztowym

ELEKTROMECHANIKA

ul. Cegielniania 17
32-410 DOBCZYCE

B-9

ZX SPECTRUM
ATARI
system turbo,
TIMEX FDD 3000,

programy użytkowe, edukacyjne, gry, instrukcje, podręczniki
wysyłka na cały kraj
rachunki
informacje po nadesłaniu koperty + znaczek.

2"P.K.T.S." Studio Komputerowe
00-103 Warszawa
ul. Królewska 43 m 25

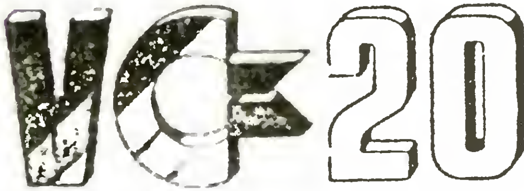
B52

Atari XE, XL, Spectrum, Timex

Niskie ceny programów na taśmie i dysku
Interfejsy Turbo I "AY" do Spectrum
Informacja — zaadresowana koperta + znaczek
05-220 Zielonka Skr. Pocz. 9/2

B 44

Programy, pamięci



tel. 20 45 28
ul. Karasia 18
64-100 Leszno

B-30

„BETA B"

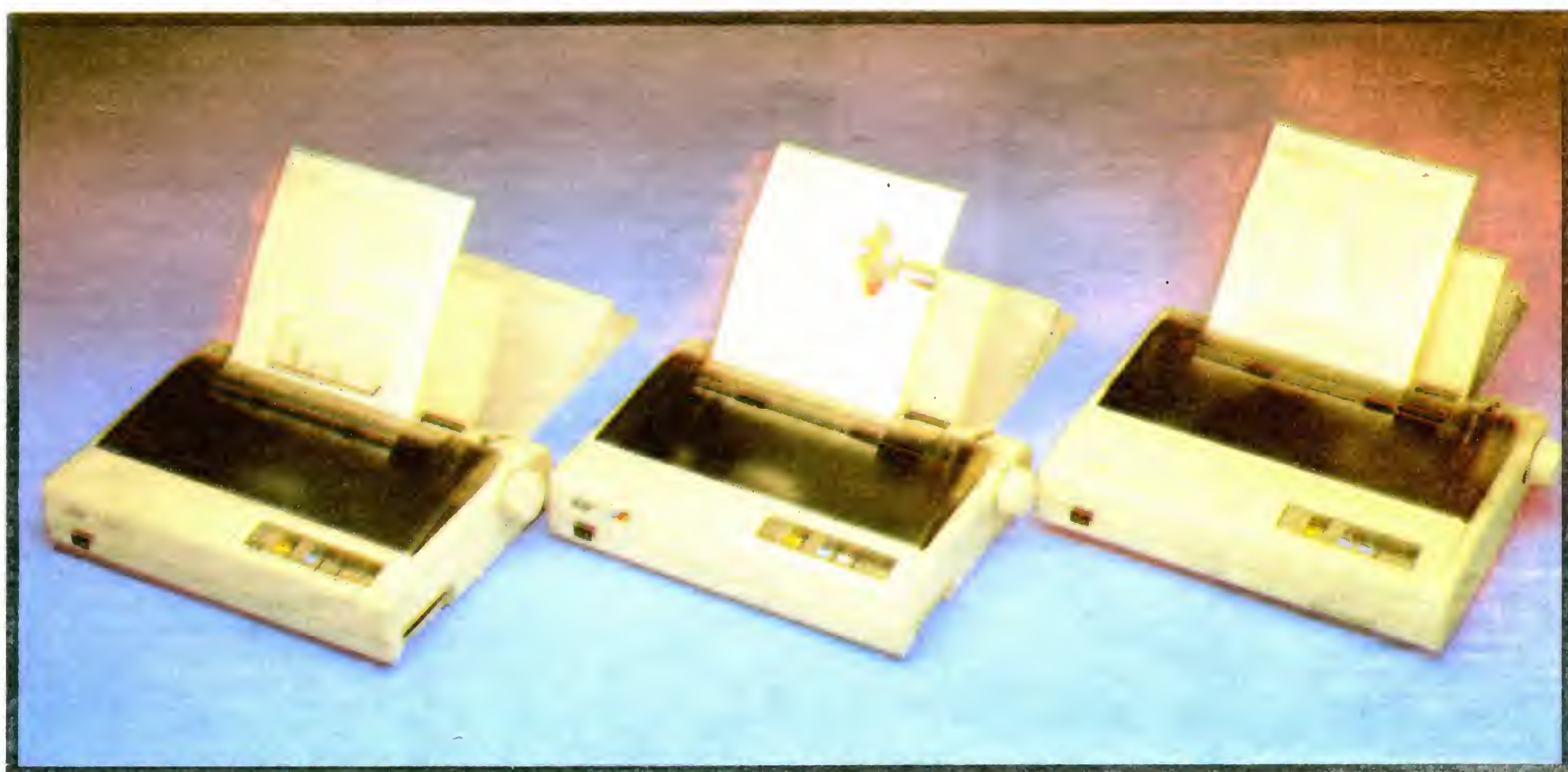
AGENCJA INFORMATYCZNA
41-200 Sosnowiec,
skrytka 254
Telef. 632-935 690-385
oferuje również wysyłkowo: Programy, Instrukcje, Literaturę dla komputerów ACORN AMSTRAD ATARI COMMODORE SHARP IBM

B 18

New New New

Drukarka Star

to najlepszy przyjaciel Twojego Komputera!



STAR LC — 10
Najpopularniejsza drukarka świata! Ponad 2.000.000 użytkowników nie może się mylić! Jeśli chcesz mieć drukarkę solidną i niezawodną, wszechstronną i niedrogą, to decyzja jest prosta: LC–10!
Prędkość druku:
144zn/s (draft)
36zn/s (NLQ)

STAR LC — 10 colour
Ta sama japońska jakość i niezawodność jak LC–10, z dodatkową możliwością druku w 7 kolorach!
Wystarczy tylko dzałożyć barwną taśmę.

Prędkość druku:
144zn/s (draft)
36zn/s (NLQ)

STAR LC24 — 10
Drukarka roku 1989 w Europie Zachodniej! Doskonała jakość pisma dzięki nowoczesnej, 24-igłowej głowicy.
Niezastąpiona do korespondencji oraz precyzyjnej grafiki.

Prędkość druku:
170zn/s (draft)
57 zn/s (LQ)

Wszystkie trzy drukarki posiadają cztery kroje czcionek, wbudowany traktor oraz funkcje „Paper Park". Można w nich stosować papier z perforacją lub bez, jak również pojedyńcze kartki.

UWAGA: W celu zainstalowania polskich znaków, prosimy skontaktować się z jednym z naszych punktów serwisowych!

Ceny:
Ceny:	LC–10	DM 350, kaseta barwiąca (czarna) DM 7,20
	LC–10 colour	DM 450, kaseta barwiąca (kolorowa) DM 14,-
	LC24–10	DM 600, kaseta barwiąca (czarna) DM 11, karta z polskimi znakami DM 100

Oto adresy naszych składów celnych i punktów serwisowych:

ABC Data CI CUP, Warszawa, ul. Żurawia 4a, tel. 21.75.08
ABC Data Service/Take, Warszawa, ul. Konopnickiej 6, tel. 28.92.81
ECS, Warszawa, ul. Połczyńska 96, tel. 36.82.50
ZIPO, Gdańsk, ul. Krynicka 1, tel. 41.82.75
Interbit, Kielce, ul. Manif. Lipcowego 4, tel. 441.99
Sykomat, Kraków, ul. Skawińska 11, tel. 21.95.40
Techmex, Bielsko Biała, ul. M.C. Skłodowskiej 13, tel. 421.98

star
the ComputerPrinter

ABC Data

Przedstawiciel na Polskę: ABC Data GmbH, ...
Bürgerstraße 12, 5300 Bonn 2. RFN.
Tel: 0228 354480/90, Tlx: 172283746, Fax: 0228 355635

KOMPUTER Z BAJERAMI



Mało który z posiadaczy komputerów w Polsce zastanawia się, w jakich warunkach powinny pracować te pożyteczne urządzenia. Wystarczy zwykłe biurko, czasem oświetlone lampą, aby kawałek klawiatury wraz z właścicielem znalazł swój kąt do pracy.
Do zbytecznych luksusów zaliczamy filtry przeciwodblaskowe, podkładki pod myszy, specjalne pokrowce — zastępowane starym ręcznikiem — czy wygodne meble „komputerowe".

Co kraj to obyczaj. Na świecie komputerowe dodatki podobnie jak w modzie, stały się dochodową gałęzią drobnego przemysłu. Gustowne fotele i krzesła dla zapracowanych programistów, meble w pastelowych kolorach, torby na duże — tak tak — komputery i setki wzorów pudełek na dyskietki to tylko część biznesu, mającego zaspokoić gusta fanów informatyki. Producenci sprzętu i oprogramowania starają się w ten sposób reklamować swój towar. Bo nic tak nie przyciąga uwagi jak koszulka z podobizną **Steve Jobsa** czy parasolki z napisem **Toshiba.** A wszystko dla *klienta.*



MEBLE I MEBELKI

Dla programisty najważniejsze jest wygodnie urządzone miejsce pracy. Musi być jasne, przestronne, z łatwym dostępem do telefonu i drukarki. Fotel lub krzesło na kółkach są obowiązkowo anatomicznie dopasowane do siedzenia właściciela. Nie bez znaczenia jest też możliwość wyciągnięcia nóg na blat biurka. W pracy obowiązuje luz.

W branży meblarskiej najlepsi są Włosi, Szwedzi i Amerykanie. To oni nadają styl. Ostatnio dobijają do nich Chińczycy z Tajwanu i Polacy z Furnelu. Ci pierwsi proponują biurka, na których mieszczą się obok siebie komputer, drukarka, telefon, pudełka na dyskietki. Ci drudzy liczą na bardziej wybrednego klienta, ale za to z grubszym portfelem. Potomkowie Leonarda i Michała Anioła dbają o oryginalność i piękno. W swych rozwiązaniach stosują lekkie ażurowe konstrukcje. Praca w takich warunkach praca staje się przyjemnością.

WYGODA I BEZPIECZENSTWO

Obok stresu największym wrogiem ludzi pracujących z komputerami jest monitor — źródło niebezpiecznego promieniowania elektromagnetycznego. Tym groźniejsze, że w przeciwieństwie do zwykłego telewizora znajduje się ono w odległości — 100 – 150 cm od nas. Pisałem w roku ubiegłym o filtrach przeciwodblaskowych i antystatycznych wykonanych ze specjalnych siatek. Dziś to już przeszłość. Kilka firm, w tym **Polaroid Corporation**, zaprezentowało w tym roku nowy rodzaj filtrów wykonanych ze specjalnego szkła. Jak zapewniał mnie **Lawrence W. Johnson Jr.** szef marketingu **Polaroida** na Europę, szkło to jest stosowane w budowie kabin nowoczesnych samolotów myśliwskich. Filtr w 98% eliminuje niebezpieczne promieniowanie elektromagnetyczne, i aż w 99% szkodliwe dla wzroku refleksy światła. Jest produkowany dla wszystkich typów monitorów komputerowych. Jedyny mankament to wysoka cena sięgająca nawet kilkuset dolarów.

Podobną funkcję pełnią ruchome podstawki na monitory, pozwalające na płynną regulację ustawienia ekranu tak pod względem wysokości, odległości, jak i kąta nachylenia względem wzroku. Są przy tym bardzo nowoczesne w kształcie. Ich zaletą jest możliwość indywidualnego ustawienia monitora przed użytkownika.



Bywa, że komputer musi pracować w zapylonej hali przemysłowej. Najlepiej jest umieścić go w specjalnym klimatyzowanym pomieszczeniu, ale jest to zbyt kosztowne. Można zatem wymienić obudowę na solidniejszą i hermetyczną. Są jednak warunki, w których to nie wystarczy. Pewna amerykańska firma proponuje klientom obudowę **„Military"**! A dotyczy to laptopów. Takie cacko zadowoli nawet Arnolda Schwarzeneggera. Całość w kolorze khaki, kuloodporna obudowa, własne zasilanie z typowych baterii wojskowych stosowanych w US ARMY i absolutna szczelność. Laptop **„Military"** wyposażony jest w specjalne złącza umożliwiające mu komunikację z innymi, typowymi komputerami wojskowymi armii amerykańskiej.

MYSZKI, SZMATKI I SZAMPONY

Pokojowo nastawione „mszy" nie znoszą brudu, nie tolerują nierównych powierzchni i lubią mieć swoje miejsce. Aby im dogodzić, wymyślono „Mouse Pad" — podkładkę ze specjalnej pianki — zabezpieczającą przed zanieczyszczeniami. Obecnie „Pad" to podstawowy dodatek każdego szanującego swoją „mysz" użytkownika. Podobnie, jak coś w rodzaju pojemnika na „myszkę", przyklejane do bocznej ścianki komputera.

Drobiny kurzu szkodzą nie tylko delikatnej naturze „myszek", lecz także klawiaturom, jednostkom centralnym i monitorom. Z pomocą przyszły firmy produkujące chemiczne środki czystości i opracowały specjalne płyny do ...mycia komputerów. Takie „komputerowe" szampony. Sprzedaje się je w zestawach przypominających walizeczki. W środku: „szampon", pojemnik przypominający z wyglądu dezodorant, a zawierający środek antystatyczny, i cały zestaw ściereczek do pucowania naszego ulubieńca. Bo o komputer trzeba dbać!

Produkcja dodatków związana jest z potrzebą ułatwienia pracy ludziom. Oferta będzie rosła, bo któż może przewidzieć, co jeszcze może być potrzebne. W Polsce niestety poza producentami filtrów przeciwrefleksyjnych i pudełek na dyskietki nie ma większego zainteresowania dla tych drobiazgów. Jest to tym bardziej niezrozumiałe, że liczba użytkowników sprzętu komputerowego rośnie. Czyżby nikomu nie zależało na przykład na ochronie ich zdrowia? Obym się mylił.

Marek Czarkowski

